



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ

FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY

INSTITUTE OF INFORMATICS

ALGORITMIZACE PRO PODPORU ROZHODOVÁNÍ

ALGORITHMIZATION FOR DECISION SUPPORT

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Juraj Kvapil

VEDOUCÍ PRÁCE

SUPERVISOR

prof. Ing. Petr Dostál, CSc.

BRNO 2020

Zadání bakalářské práce

Ústav: Ústav informatiky
Student: **Juraj Kvapil**
Studijní program: Systémové inženýrství a informatika
Studijní obor: Manažerská informatika
Vedoucí práce: **prof. Ing. Petr Dostál, CSc.**
Akademický rok: 2019/20

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává bakalářskou práci s názvem:

Algoritmizace pro podporu rozhodování

Charakteristika problematiky úkolu:

Úvod
Cíle práce, metody a postupy zpracování
Teoretická východiska práce
Analýza současného stavu
Návrh řešení a přínos návrhů řešení
Závěr
Seznam použité literatury
Přílohy

Cíle, kterých má být dosaženo:

Diplomová práce se zabývá obchodováním s kryptoměnami a nástroji pro podporu rozhodování. Výstupem práce bude vytvoření software pro predikci vývoje cen vybraných kryptoměn na základě technické analýzy. Pro tvorbu software bude využito specializovaných nástrojů, např. MultiCharts, do kterého budou implementována data z kryptoměnových burz.

Základní literární prameny:

DOSTÁL, P. Advanced Decision Making in Business and Public Services. Brno: CERM, 2011. 168 s. ISBN 978-80-7204-747-5.

DOSTÁL, P. Pokročilé metody rozhodování v podnikatelství a veřejné správě. Brno: CERM, 2012. 718 s. ISBN 978-80-7204-798-7.

GRAHAM, B. Inteligentní investor. Praha: GRADA, 2007. 504 s. ISBN 978-80-247-1792-0.

VIGNA, P. and M. CASEY. The age of cryptocurrency: how bitcoin and digital money are challenging the global economic order. New York: St. Martin's Press, 2015. ISBN 978-1-250-06563-6.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2019/20

V Brně dne 29.2.2020

L. S.

doc. RNDr. Bedřich Půža, CSc.
ředitel

doc. Ing. et Ing. Stanislav Škapa, Ph.D.
děkan

Abstrakt

Bakalárska práca sa zaoberá vývojom v oblasti automatizácie elektronického obchodovania. V práci je demonštrované využitie automatického obchodného systému, ktorý obchoduje na základe technickej analýzy. Tento systém sa zameriava konkrétne na oblasť rozpoznávania vzorov v grafoch. V práci je testovaný na kryptomene Bitcoin, ako aj na menovom páre EUR/USD. Jeho využiteľnosť teda nekončí len pri kryptomenách, môže byť nasadený na ľubovoľný inštrument. Jeho cieľom je zlepšenie obchodníckej schopnosti reagovať a to prostredníctvom automatizácie niektorých analytických a obchodných úkonov.

Kľúčové slová

obchodovanie, stratégie, kryptomeny, burza, analýza, vzory, automatizácia, bitcoin

Abstract

The bachelor thesis is focused on area of algorithmic trading. The main focus in this thesis is aimed towards algorithmic solution to technical analysis on cryptocurrencies such as Bitcoin as well as on other instruments, for example currency pair EUR/USD. You will be presented with my method to maximize productivity and make trading effortless. My solution is based on real time running software, that is capable of detecting predetermined patterns in price charts. Software has also ability to place automated trading orders based on results of analysis.

Key words

trading, strategy, cryptocurrencies, forex, analysis, patterns, automation, bitcoin

Bibliografická citácia

KVAPIL, Juraj. *Algoritmizace pro podporu rozhodování* [online]. Brno, 2020 [cit. 2020-05-17]. Dostupné z: <https://www.vutbr.cz/studenti/zav-prace/detail/125519>. Bakalářská práce. Vysoké učení technické v Brně, Fakulta podnikatelská, Ústav informatiky. Vedoucí práce Petr Dostál.

Čestné prehlásenie

Prehlasujem, že predložená bakalárska práca je pôvodná a spracoval som ju samostatne.
Prehlasujem, že citácie použitých prameňov sú úplné, že som vo svojej práci neporušil autorské práva (v zmysle Zákona č. 121/2000 Sb., o práve autorskom a o právach súvisiacich s právom autorským).

V Brně dňa 17.5 2020

.....

podpis autora

Pod'akovanie

Týmto ďakujem svojmu vedúcemu bakalárskej práce prof. Ing. Petr Dostál, CSc. za poskytnutie cenných rád, odbornej pomoci, vecných pripomienok a za jeho poskytnutý čas pri konzultáciach.

OBSAH

ÚVOD.....	10
CIELE PRÁCE, METÓDY A POSTUPY SPRACOVANIA	11
1 TEORETICKÉ VÝCHODISKÁ PRÁCE	12
1.1 Úvod do teórie finančných trhov.....	12
1.2 Historický vývoj používania peňazí.....	13
1.3 Čo je to forex.....	14
1.4 Ako sú interpretované cenové kurzy	17
1.5 Ako vyzerá obchodovanie na burze	19
1.6 Čo sú to kryptomeny	25
1.6.1 Čo je to Bitcoin	26
1.7 Ako obchodovať s kryptomenami.....	28
1.8 Čo sú to technické indikátory.....	28
1.8.1 Jednoduchý klzavý priemer SMA.....	29
1.8.2 Bollingerové pásmo	30
1.8.3 Oscilátory.....	31
1.9 Analýzy	34
1.9.1 Psychologická analýza.....	34
1.9.2 Význam fundamentálnej analýzy.....	35
1.9.3 Význam technickej analýzy	36
2 ANALÝZA SÚČASNÉHO STAVU	39

2.1	Dostupné platformy pre algorimické obchodovanie	40
2.1.1	Quantopian.....	40
2.1.2	Quantconnect	41
2.1.3	MetaTrader 4.....	42
3	VLASTNÉ NÁVRHY RIEŠENIA.....	44
3.1	Princíp definovania Patternu	45
3.2	Princíp fungovania algoritmu.....	47
3.3	Implementácia algoritmu	52
3.3.1	Implementácia podporného indikátoru	54
3.3.2	Definícia patternu v kóde.....	57
3.3.3	Priebeh procesu identifikácie patternu.....	58
3.3.4	Zadávanie obchodných príkazov	61
3.4	Demonštrácia algoritmu	62
3.5	Vyhodnotenie	67
3.5.1	Užívateľská prívetivosť	67
3.5.2	Robustnosť kódu.....	68
3.5.3	Ekonomické zhodnotenie.....	68
	ZÁVER	70
	ZOZNAM POUŽITÝCH ZDROJOV	71
	ZOZNAM POUŽITÝCH OBRÁZKOV	72

ÚVOD

Dostupnosť výpočtovej techniky, ako sú počítače, smartfóny alebo tablety v dnešnej dobe otvárajú cestu, kedysi nekonvenčným, ba až nepredstaviteľným možnostiam prístupu na otvorený trh.

Táto práca sa zameriava v prvom rade na možnosť využitia výpočtovej techniky ako jednotlivcom, tak aj komerčnými spoločnosťami, ktoré hľadajú možnosti efektívneho spravovania voľného kapitálu. Predostiera možnosti použitia algoritmov v prospech obchodníka.

Dnes môže, aj malý obchodník pristupovať k mnohým inštrumentom do ktorých je možné investovať svoj voľný kapitál a s trochou šťastia ho aj efektívne zhodnotiť. Ja som sa rozhodol zamerať na kryptomeny. Kryptomeny sú veľmi lákavou komoditou, kvôli svojej vysokej volatilite a možnosťou vysokých zárobkov. Na druhej strane sú vysoko rizikovou investíciou.

Na základe vysokej dostupnosti vzdelávacích materiálov, návodov či odborných článkov na internete, v kombinácii s dostupnosťou pokročilých vývojárskych a analytických nástrojov, si užívateľ dokáže zostrojiť vlastné algoritmy ku svojim stratégiám a vytvoriť si svoje obchodné portfólio s ohľadom na vlastnú únosnosť rizík a to bez potreby portfólio manažéra. Obchodník si dokáže jednoduchšie, než kedykoľvek predtým vytvoriť funkčné portfólio rovno z pohodlia domova.

CIELE PRÁCE, METÓDY A POSTUPY SPRACOVANIA

Práca sa zaoberá automatickými analytickými nástrojmi, ktoré by mali umožniť obchodníkovi reagovať včas na aktuálnu situáciu trhu, alebo ho v reálnom čase varovať o udalostiach, ktoré prináša otvorený trh.

Cieľom tejto práce je prezentácia potenciálu automatických obchodných systémov prostredníctvom vytvorenia softvéru, pomáhajúceho s obchodovaním na základe technických analýz.

Posúdenie dostupných existujúcich nástrojov, ako algoritmických, tak aj priamo obchodných určených pre bežných ľudí. Praktickou aplikáciou týchto nástrojov prísť k nadobudnutiu cenných strategických vedomostí ohľadom sledovaného inštrumentu, úspore času pri obchodovaní, prípadne priamo k finančnému zisku.

Cieľom je teda vytvoriť a predstaviť nápomocný nástroj k technickým analýzam, ktorý zvýši reakčnú schopnosť obchodníka na dáta z trhu a to pomocou automatizovaných analýz cenových grafov. Tieto analýzy sa zameriavajú na rozpoznávanie opakujúcich sa vzorov v grafoch. Následne na základe týchto analýz nástroj poskytuje možnosť automaticky generovaných obchodných príkazov.

1 TEORETICKÉ VÝCHODISKÁ PRÁCE

Táto časť bakalárskej práce sa venuje teoretickým východiskám spojeným s danou problematikou. Obsahuje teóriu nutnú k porozumeniu problematike a princípu riešenia, ktoré vedie k dosiahnutiu stanovených cieľov.

1.1 Úvod do teórie finančných trhov

V každej ekonomike sa bežne stáva, že sa ekonomické subjekty dostávajú do situácií, kedy majú buď prebytek, alebo naopak nedostatok peněžních prostředků, který jim brání v realizaci jimi zamýšlených činností. Někdo např. vymyslí vynález, který by mohl mít pro společnost značný význam. Nemá však prostředky na to, aby jej zrealizoval. Jiný naopak disponuje nevyužitými penězi. Pokud se podaří tyto dva subjekty vzájemně propojit, dojde k zavedení uvedeného vynálezu do praxe a vynálezce, poskytovatel peněz i ekonomika si polepší. (2, s. 32)

Stejně jako jednotlivci mohou peněžní prostředky potřebovat i firmy nebo stát. Firma může potřebovat peníze např. na modernizaci výroby nebo na umístění svých nových výrobků na trh; místní správa na výstavbu nové nemocnice, na opravu komunikace nebo školy; vláda na výstavbu vojenského letiště nebo na odstraňování následků vzniklých v důsledku přírodní katastrofy. (2, s. 32)

Peněžní prostředky tedy mohou potřebovat všechny druhy ekonomických subjektů, a to jak domácnosti, firmy, tak i stát. A je posláním finančních trhů, aby přemísťovaly peníze od přebytkových subjektů (které jich mají v dané době nadbytek) k subjektům deficitním (které si naopak přejí utratit více, než kolika penězi disponují), a to především k těm, jež pro ně mají nejefektivnější využití. (2, s. 32)

A protože finanční trh nemůže fungovat samostatně, nýbrž pouze v rámci celého finančního systému, jež je zase nedílnou součástí ekonomiky, nelze teorii finančního trhu chápat odděleně ani od jiných oblastí finanční teorie, ani od ekonomie jako vědní disciplíny, za jejíž součást je možno finanční teorii – nazývanou též „peněžní ekonomie“ – považovat. (2, s. 32)

1.2 Historický vývoj používania peňazí

Je zřejmé, že jak fungování finančního trhu, tak i činnost celého finančního systému jsou podmíněny existencí peněz. To vyplývá jednak z toho, že v moderní ekonomice je peněžní směna nutným předpokladem fungování všech existujících druhů trhů, jednak z toho důvodu, že v rámci finančního systému zabezpečují peníze ještě i další funkce, kterých postupně nabyly v průběhu svého vývoje. A to je také důvod, proč je zapotřebí věnovat jejich vývoji alespoň krátce pozornost. (2, s. 32)

Ekonomická teorie považuje peníze za jeden z největších objevů lidstva. Tento názor vychází především z toho, že zavedení peněz vyřešilo problémy s naturální směnou (jež vyžadovala tzv. oboustrannou shodu potřeb). Tu nahradila směna peněžní, jež (protože peníze fungují jako prostředek směny) oboustrannou shodu potřeb nevyžadují). Přitom je zřejmé, že podmínkou toho, aby mohlo „něco“ vykonávat tuto univerzální funkci bylo, že toto „něco“ museli být všichni lidé ochotni přijímat, což bylo podmíněno tím, že si jsou jisti, že „to“ kdykoli přijme každý jiný člověk. (2, s. 32)

Zpočátku byla důvěra lidí v peníze založena na tom, že jejich funkci plnily drahé kovy (především zlato a stříbro), které mezi lidmi obíhaly fyzicky. Podstatou této důvěry bylo všeobecné vědomí, že drahé kovy jsou dlouhodobě vzácné a homogenní, na což brzy logicky navázala ražba zlatých a stříbrných mincí. Nicméně se později ukázalo, že podstatou peněz není, zda jsou vyrobeny z drahých kovů či nikoli, nýbrž, že rozhodující je důvěra lidí v to, že je za drahé kovy mohou kdykoli bezproblémově vyměnit. To se prokázalo zavedením papírových peněz směnitelných za zlato, jež zpočátku emitovaly soukromé banky, na což později navázalo prohlášení peněz (bankovek a mincí) za zákonné platidlo, v důsledku čehož je již nebylo možné nikým odmítnout. Tím se stalo, že důvěra v peníze začala souviset s důvěrou lidí ve státní moc. (2, s. 32)

Další etapou ve vývoji finančního systému bylo postupné rozšiřování bankovních služeb včetně zavádění šeků (a posléze platebních karet). To umožnilo lidem disponovat s penězi uloženými na bankovních účtech i jinak, než pouze prostřednictvím hotovostních peněz, a tudíž vlastně vznikly první finanční – v tomto případě platební instrumenty. Ty na jedné straně umožnily lidem bezhotovostně platit, zároveň však způsobily, že jejichž

prostřednictvím prováděné platby si banky musely mezi sebou vzájemně vyúčtovávat a tudíž začít realizovat mezibankovní převody peněz „z účtu na účet“. (2, s. 32)

V době naturální směny byla vzájemná výměna dvou produktů jediným obchodním aktem. Při obchodní výměněs využitím peněz tomu tak už ovšem nebylo, protože tím, že se peníze (jako prostředek směny) začaly včleňovat mezi obě vzájemně směňovaná zboží, tak se akt souběžného prodeje a koupě zpravidla rozdělil na dva, časově odlišné směnné akty. Z toho vyplývá, že ekonomické subjekty začaly (i když zpočátku většinou pouze dočasně) disponovat určitou „peněžní hotovostí“, kterou mohly držet buď ve formě oběživa (držbou bankovek a mincí), nebo ji mohly mít uloženu na bankovních účtech. Jinými slovy se stalo, že lidé (resp. ekonomické subjekty) začaly vlastnit různá množství peněz, které začaly tvořit součást jejich „majetkových aktiv“ (bohatství). Vedle peněžních zůstatků ovšem vlastnili i jiná majetková aktiva, jako např. pozemky, domy, resp. různé movité věci, ke kterým posléze přibýly i různé druhy tzv. finančních investičních instrumentů (akcie, dluhopisy, apod.). Všechna tato majetková aktiva představují bohatství jejich majitelů, nicméně pokud se týče jejich vlastností, tak těmi se mohou podstatně lišit. Některá z nich (tzv. investiční aktiva), přinášení svým vlastníkům výnosy, ale může být obtížné je zpětně směnit (prodat na finančním trhu) za hotové peníze. Jiná jsou naopak likvidní, avšak žádné významné výnosy svým majitelům nepřinášejí, přičemž nejlikvidnějším existujícím majetkovým aktivem jsou právě peníze. A protože jsou vysoce likvidní a lze je tudíž kdykoli směnit za kterékoli „zboží“, existuje po nich z tohoto důvodu pochopitelně poptávka (tzv. poptávka po peněžních zůstatcích). A protože na finančních trzích jsou tímto „zbožím“ různé druhy tzv. finančních investičních nástrojů, jejichž nákupem majitelé peněžních zůstatků své peníze investují, čímž je mění ve finanční kapitál (nesoucí výnos), pomáhají financovat činnost celého ekonomického systému, jehož jsou příslušný finanční trh a potažmo i celý finanční systém nedílnou součástí. (2, s. 33)

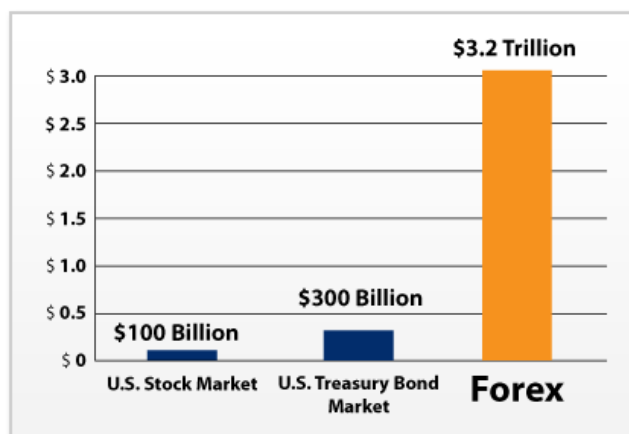
1.3 Čo je to forex

FOREX = Internatioal Interbank FOReign EXchange. Zjednodušene obchodovanie s cudzími menami alebo aj devízový trh. (3, s. 17)

Jedná sa o obchodný systém sprostredkujúci výmenu mien, takzvaných hlavných a vedľajších menových párov, stredné kurzy hlavných svetových búrz sa považujú oficiálne menové kurzy. Jedná sa o najlikvidnejší trh na svete, jeho objemy sa pohybujú v biliónoch dolárov každý obchodný deň. (3)

Hodnoty jednotlivých kurzov reflektujú relatívnu silu ekonomík daných krajín ale aj odzrkadľujú záujmy centrálnych bánk o riadenie monetárnej politiky v prospech štátu. Vstup na forex je podobný ako u akciových trhov a teda sa s menovými párami pri obchodovaní narábame podobne ako s akciami, teda obchodujeme s predpokladom rastu alebo poklesu budúceho kurzu prostredníctvom nákupov alebo predajov. (3)

Na nasledujúcom obrázku možno vidieť porovnanie denných objemov forexu s inými odvetviami.



Obrázok č. 1: Porovnanie objemov na forexe s inými trhami

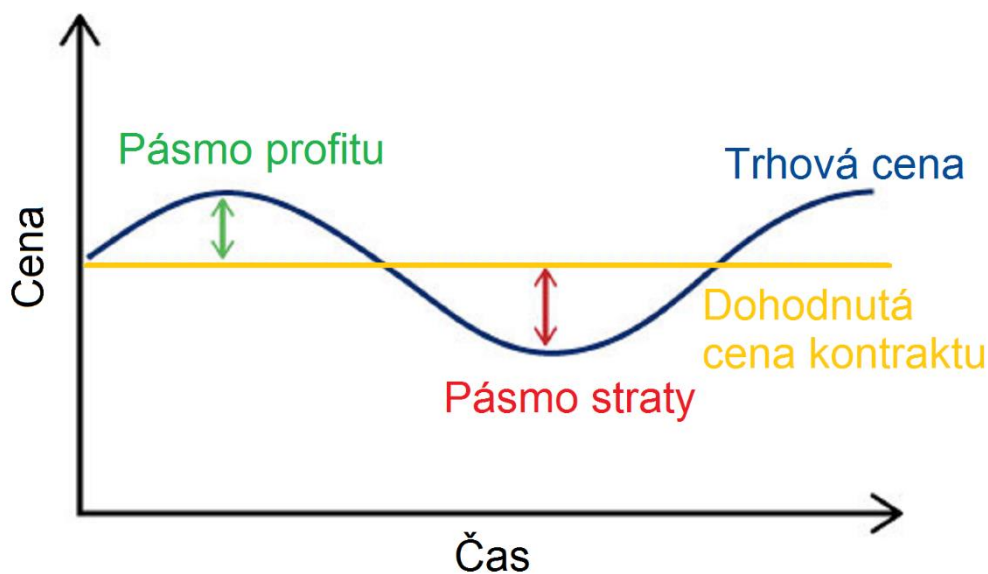
(Zdroj: Vlastné spracovanie podľa: 4)

Počiatky forexu tak ako ho poznáme dnes sa datujú do začiatku 70-tých rokov kedy sa uviedli do prevádzky prvé systémy umožňujúce bankám a veľkým investorom výmenu svojich aktív. Môžeme tiež povedať že vtedy nadobro skončila doba mien krytých zlatými rezervami. Rozmachom internetu pár rokov po miléniu sa k burzám mohli pripájať aj menší obchodníci a na živo sledovať vývoje cien, do tejto doby objednávky prebiehali často telefonicky na ústredne kde odbavovali. (12)

Dnes sa ľudovo za forex nepovažujú len menové páry ale celkovo prístup na burzu s menami, akciami a komoditami ktoré možno obchodovať pod rôznymi druhmi kontraktov (termínované alebo promptné). Veľmi často sa však stretávame práve

s termínovanými (hlavne teda Futures) ktoré nám umožňujú obchodovať na páku. Pákové obchodovanie v praxi znamená že obchodník môže znásobiť svoju kúpnu silu, v niektorých prípadoch až 100-násobne. To však nesie riziko toho, že sám svojím kapitálom nemusí vedieť pokryť stratu z obchodu s tak veľkým objemom. Termínované kontrakty možno interpretovať ako dohody o budúcich obchodoch kde sa obe strany prisľubujú k fixným objemom za fixné ceny, to nám dovoľuje pákovo obchodovať aj keď v čase otvorenia kontraktu nedisponujeme daným kapitálom. Po otvorení tohto kontraktu musíme sledovať ďalej sa pohybujúci kurz aktíva a sledovať rozdiel medzi dohodnutou cenou a cenou aktuálnou, ak sa aktuálna cena pohybuje nevýhodným smerom je rozumné obchod zavrieť, než prerobíme viac finančných prostriedkov na účte než máme k dispozícii. Zavretie technicky znamená odpredanie svojho záväzku niekomu inému a teda neznamená zrušenie celej dohody o budúcom obchode. Na nasledujúcom obrázku možno vidieť vývoj hodnoty futures kontraktu v závislosti na tržnej cene podkladového aktíva.(12)

CENA FUTURES KONTRAKTOV



Obrázok č. 2: Vývoj hodnoty futures kontraktu

(Zdroj: Vlastné spracovanie)

1.4 Ako sú interpretované cenové kurzy

Obchodníci pri vytváraní investičných rozhodnutí zvyknú upriamovať zrak k historickým vývojom kurzov. Kurzy sa pri obchodovaní na burze zvyknú zobrazovať vo forme takzvaných sviečkových grafov, tie sa skladajú z množstva dvojíc vertikálnych čiar, jednej hrubšej a jednej tenšej kde každá dvojica opisuje vývoj ceny za istý predom zvolený časový rámec, obvykle 1/5/15/30/60/240 minút alebo 1/7/30 dní, pričom v praxi, pri technickej analýze, sa najviac zvyknú používať práve hodinové alebo denné časové rámce, sledovanie menších časových rámcov než hodinových, sa nezvykne robiť, pokiaľ sa nesnažíme obchodovať na malých pohyboch kurzu, kde však začínajú hrať rolu v našom rozhodovaní aj „spready“ (poplatky za obchody), ktoré sa v čase menia, a to na základe volatility danej akcie alebo komodity. Malé časové rámce sa tiež nemusia vyplatiť sledovať ľuďom, ktorých nepriaznivý vývoj ceny môže stresovať, keď že na malých časových rámcoch môžeme pozorovať výraznejšie výkyvy oproti tým väčším. (11)

Na obrázku číslo tri možno pozorovať striedajúce sa zelené a červené sviečky, kde každá predstavuje vývoj kurzu (menový pár EUR/USD) za jeden deň. Teda u každej sviečky, tenšia čiara ukazuje maximálny výkyv kurzu daného dňa a hrubá časť takzvaný otvárací a zatvárací kurz (cena na začiatku dňa a cena na jeho konci). (2)

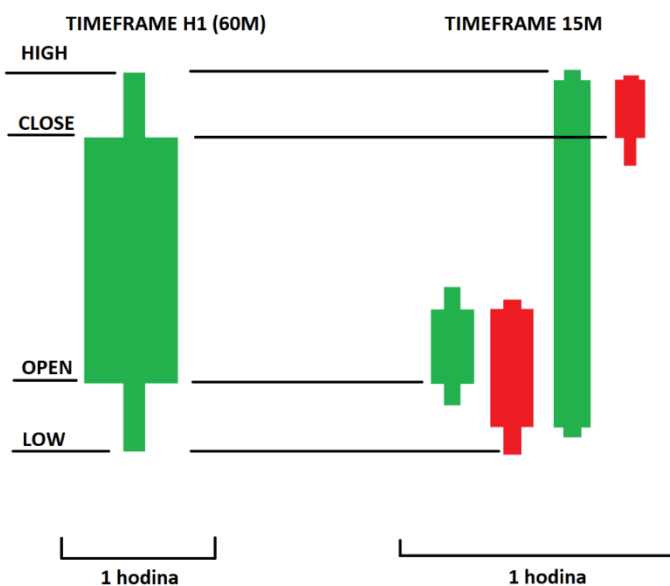


Obrázok č. 3: Sviečkový graf ukážka

(Zdroj: Vlastné spracovanie)

Na obrázku číslo štyri, máte detailne vysvetlenú kompozíciu rastúcej sviečky. Uvedený príklad ako sa líšia dva rozličné časové rámce. Môžete pozorovať, že pri využití hodinového časového rámca strácame informácie o tom, ako sa presne cena vyvíjala v rámci minút. Teda možno tvrdiť, že tým prichádzame o časť príbehu. Týmto spôsobom však vieme do grafu vtesnať dáta za dlhší čas. Pokiaľ by ste sa zaujímali o dianie v rámci jednej hodiny, je teda výhodné prepnúť na menší časový rámec (15M/5M/1M), kde môžete pozorovať menšie výkyvy trhu a aplikovať u nich svoje znalosti. (2)

V tomto prípade možno pozorovať, že prvé dve 15 minútové sviečky, sa cena držala relatívne nízko a až medzi tridsiatou a štyridsiatou piatou minútou, došlo práve k výraznému rastu. Práve toto, by sme však z hodinovej sviečky nikdy nezistili. Menší časový rámec nám taktiež umožňuje presnejšie monitorovať fundamentálne dáta a ich dopad na kurzy, môžeme sledovať pohyby v presnú minútu zverejnenia dôležitých informácií.



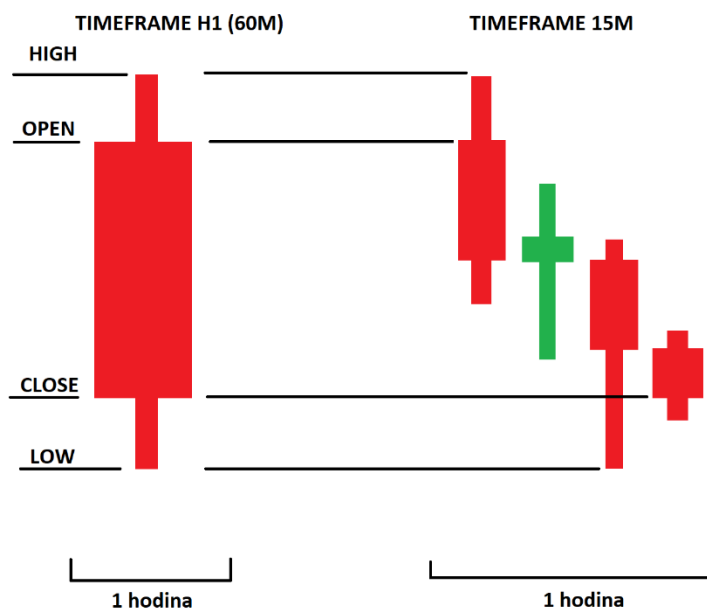
Obrázok č. 4: Kompozícia a význam rastúcej sviečky

(Zdroj: Vlastné spracovanie)

Všeobecne pomocou sviečkových grafov vieme interpretovať takmer akúkoľvek časovú radu alebo iný, v čase kontinuálny dátový tok, ktorý možno kvantifikovať. Najčastejšie, sa sním stretneme práve na burze, kde nás zaujíma práve to, ako sa cena zmenila za daný deň, a jak nestála, bola obchodná seansa v daný deň. Toto nám presne znázorňuje sviečkový graf, tieto grafy majú viacero variácií na ktoré môžeme naraziť. (2)

Môžu sa jednať len o grafické zmeny, ako sú iné farebné kombinácie alebo drobný re-dizajn sviečok, ktorý ale každý pochopí. Za zmienku však stojí sviečkový graf, ktorý zobrazuje aj objem ktorý sa v daný čas obchodoval, a to tým, že samotnú sviečku rozťahne do šírky, teda v praxi to znamená že dlhé sviečky sú často, zároveň aj široké, pretože k výraznejšiemu pohybu kurzu je potrebné aby sa zobchodoval väčší objem. (2)

Na obrázku číslo päť možno pozorovať klesajúcu hodinovú sviečku. Po jej rozanalyzovaní na 15 minútovom časovom rámci možno pozorovať vcelku jasný pokles bez významných korekcií. Možno teda usúdiť, že na trhu bol konsenzus ohľadom smerovania kurzu. Opäť, menší časový rámec ukazuje viac informácií o priebehu kurzu v danej hodine. Mnohokrát sa môže jednať o významné strategické informácie. (2)



Obrázok č. 5: Kompozícia a význam klesajúcej sviečky

(Zdroj: Vlastné spracovanie)

1.5 Ako vyzerá obchodovanie na burze

Na burze môžeme obchodovať mnoho inštrumentov, od menových párov, cez akcie, až po komodity, nevynímajúc kryptomeny, ako samostatnú kategóriu. Predtým, než možno začať obchodovať, je potreba získať prístup na túto burzu. Pokiaľ nie sme veľká investičná firma, ale jednotlivec, robí sa to prostredníctvom založenia účtu u „brokera“. Ten vám následne umožní prístup k trhu. Vo väčšine prípadov stačí predložiť občiansky

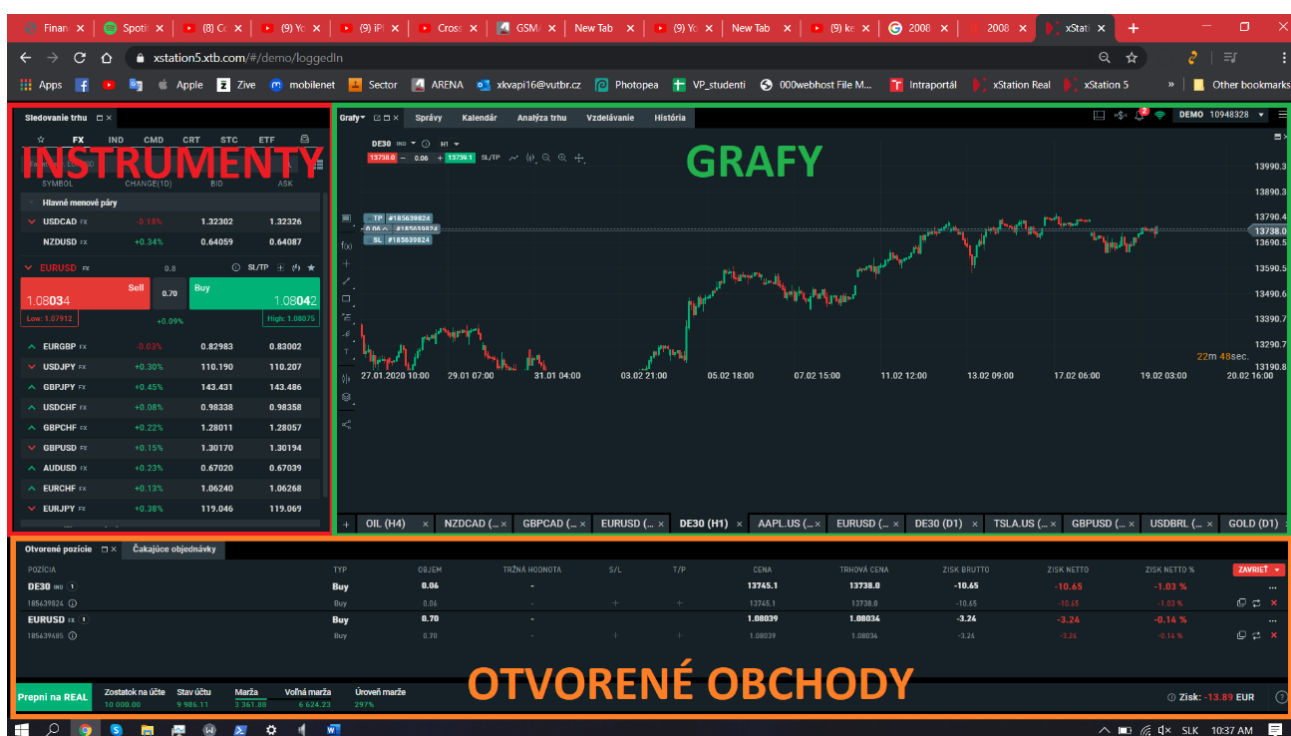
preukaz pri registrácii, pokiaľ si však zvolíte účet s možným prečerpaním, treba mať na pamäti, že v obchodných podmienkach schvaľujete aj prípadné exekučné konanie voči vám. Ak si však zvolíte štandardný účet, tak broker vám pri veľkej strate automaticky uzavrie obchody, aby ste sa nedostali do mínusu. Veľká strata je tiež sprevádzaná termínom „margin call“, kedy nám broker oznamuje, že máme nedostatok kapitálu pre udržanie pozícií bez toho, aby náš účet šiel do mínusu. Pre pochopenie výrazu slova „margin“ postačí jednoduché vysvetlenie, ak obchodujeme na páku, znamená to, že máme väčšiu kúpnu silu, než predstavujú finančné prostriedky na našom obchodnom účte. To znamená, že ak máte 1000\$ na účte a máme 10-násobnú páku, môžeme si dovoliť otvoriť obchod o hodnote 10 000\$. Avšak broker by nebol rád, keby ručí za tak veľký obchod sám, preto vašich 1000\$ zablokuje (zamaržuje). Z nich vykryje prípadnú stratu, a to práve do výšky zamaržovaných financií. V prípade zisku sa po zavretí obchodu tieto prostriedky odblokujú a pričíta sa vám aj zisk. Rizikom pri takýchto obchodoch je samotná páka, ktorá nám dokáže veľmi rýchlo vynulovať účet a to tým princípom, že ak by sa váš obchod za 10 000\$ pohol o 10% do červených čísel, stane sa to, že strácate celých 1000\$ ktoré ste vyhradili ako krytie. Tento efekt nabera na rizikovosti s tým ako veľkú páku volíte, pričom ak by ste v tomto prípade nakupovali bez páky, respektíve s pákou 1-násobnou, 10%-ná strata by v tomto prípade predstavovala iba 100\$. Samozrejme, pri prípadnom 10%-nom zisku, by ste tiež zarobili len 100\$, na rozdiel od 1000\$ pri 10-násobnej páke. Preto je pákové obchodovanie veľmi atraktívne, najmä pre malých investorov, ktorí za víziou veľkých ziskov a zároveň za relatívne nízky kapitálový výdaj, investujú na burze. (11)

Po tom ako si založíte účet u brokera, sa stretnete s jeho obchodnou platformou, dnes sa najčastejšie pre prístup používajú webové rozhrania. Ich výhodou je najmä kompatibilita, bez nutnosti sťahovania a inštalácie dodatočného software. Mnoho brokerov tiež disponuje aplikáciou pre smartfóny a tablety s veľmi podobným prostredím, občas však so zredukovanou funkcionalitou. Populárnou webovou platformou je napríklad xStation, ktorú môžete vidieť na obrázkoch nižšie. (11)

Na obrázku č. 6 môžete pozorovať, vcelku prehľadné, rozdelenie prvkov prostredia (UI). Červene vyznačená oblasť zhŕňa obchodovateľné inštrumenty do kategórií.

Pod kategóriami nájdete forex, komodity, akcie, fondy(ETF),... Tiež si môžete všimnúť dva druhy cien pri ilustrovaných menových pároch. Cena v stĺpci „**BID**“ alebo v stĺpci „**ASK**“. Volí sa medzi nimi podľa toho či sa chystáte nakupovať (BUY) alebo predávať (SELL), pričom pri nákupe, si všímate cenu v stĺpci „**ASK**“ a pri predaji cenu „**BID**“.

V zeleno označenej časti, sa nachádzajú editovateľné záložky s grafmi, U nich si môžete meniť periódy, pridávať čiary, pridávať indikátory a posúvať sa v histórii. Taktiež vidno cenové úrovne na ktorých sme vstúpili do obchodov. V spodnej časti vyznačenej oranžovo sa nachádza zoznam obchodov, ktoré sú buď otvorené alebo čakajúce na exekúciu, vidieť môžete tiež náhľad na ich zisk alebo prípadnú stratu.

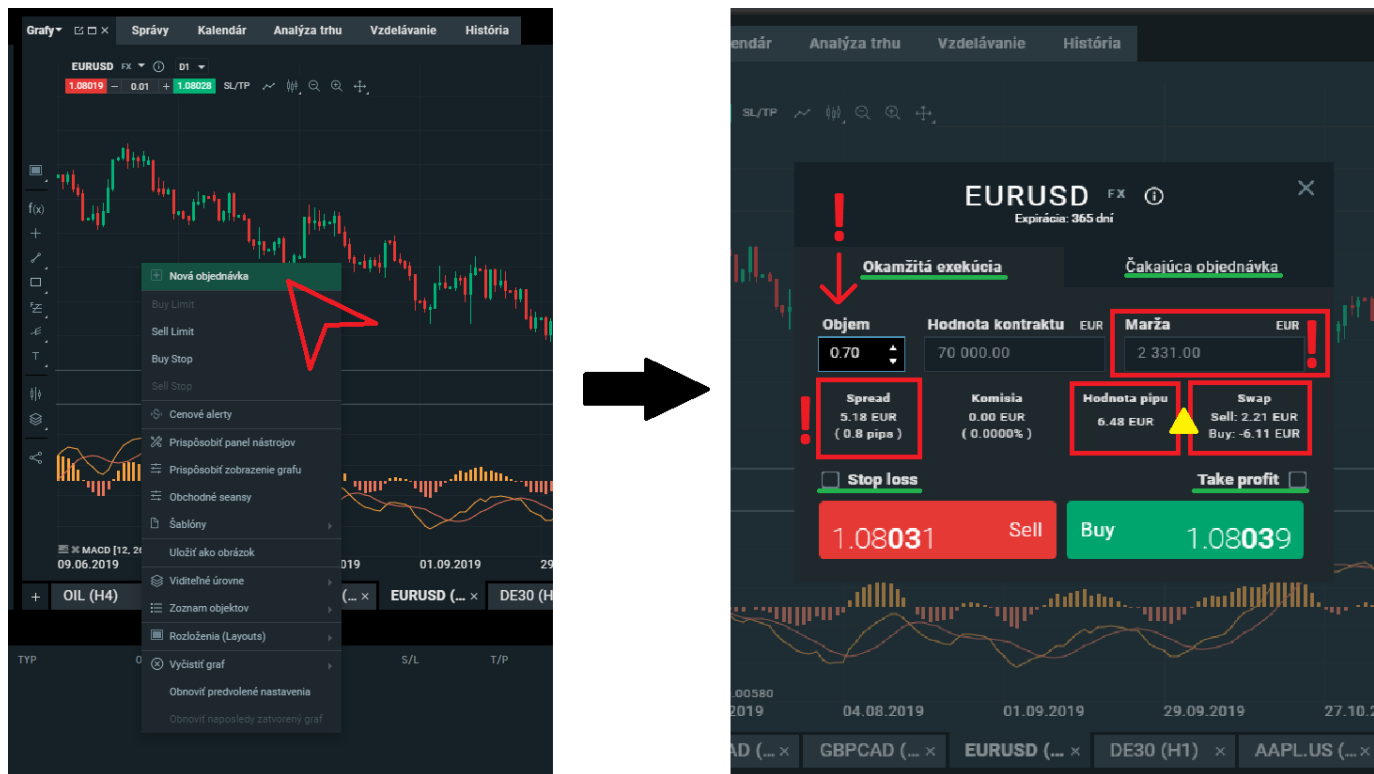


Obrázok č. 6 Rozhranie obchodnej platformy xStation 5

(Zdroj: Vlastné spracovanie)

V Obrázku číslo 7 si možno všimnúť ponuku možností, ktorá sa vyobrazí kliknutím pravého tlačidla myši do grafu. V ponuke následne možno vidieť položku menom „Nová objednávka“, ktorej zvolením, sa objaví okno k novej objednávke, vzťahujúce sa k instrumentu zobrazovanom na grafe. Objednávku možno otvoriť aj zo záložky instrumentov z predošlého obrázku.

Čo si však všimnúť pri zadávaní tohto príkazu je určite objem našej investície (v lotoch). Podľa neho, sa nám prepočíta peňažný objem a marža jemu prislúchajúca. Hodnota lotu sa líši podľa toho aký inštrument si zvolíte, teda za lot nemožno považovať istú fixnú čiastku naprieč všetkými inštrumentami.



Obrázok č. 7: Proces otvárania novej objednávky v xStation 5

(Zdroj: Vlastné spracovanie)

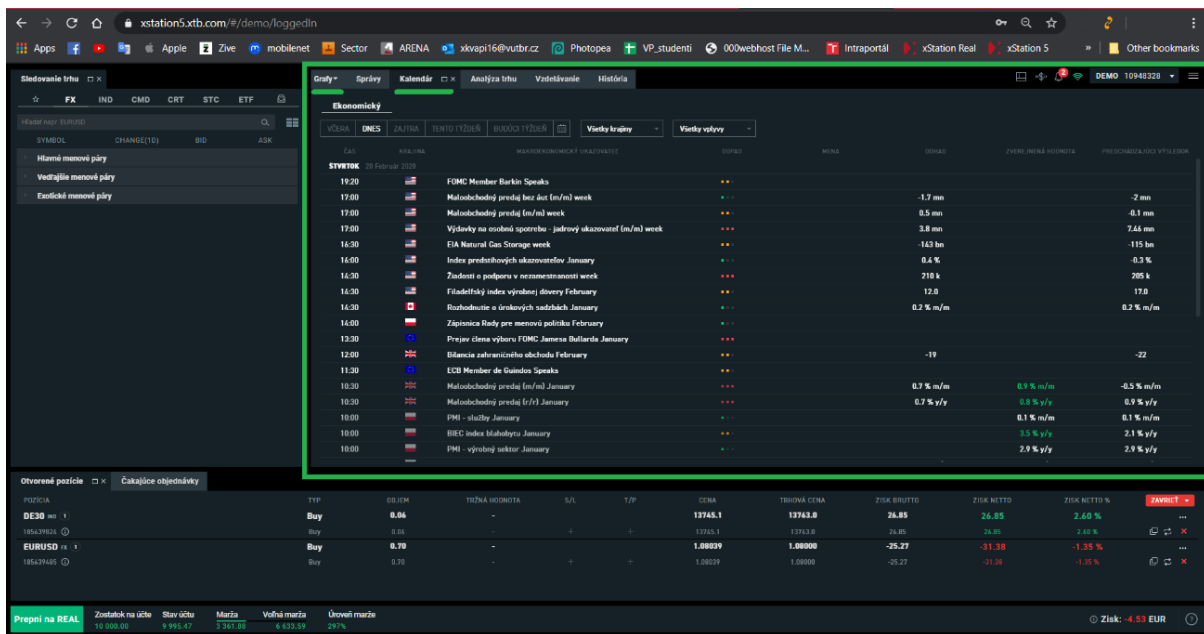
Ďalej si určite treba všimnúť to, za akú cenu nakupujete a teda či chcete otvoriť okamžitý obchod alebo na nejakej inej cenovej hladine, kam predpokladáte, že sa cena môže dostať. Vtedy sa volí čakajúca objednávka (vyznačenú zelene), kde si nastavíte cenovú hladinu, a prípadne expiráciu príkazu (ak to platforma dovoľuje). Na čo ďalšie treba myslieť, je takzvaný „spread“. V praxi, pre obchodníka znamená jednorazový poplatok brokerovi. Tieto poplatky sa dokážu pohybovať v drobných, ale aj vo väčších číslach. Je častou chybou začiatocníkov, že si tento parameter nevšimnú, a ak obchodujú malé objemy, tak im môže urobiť značný škrt cez rozpočet. Spread možno chápať aj ako rozdiel medzi nákupnou a predajnou cenou, kedy na príklad kurz pre nákup je 1.05 a kurz pre predaj 1.00, to znamená, že ak by ste nakúpili a hneď predali (tak aby sa kurz nestihol zmeniť), tak rozdiel medzi nákupným a predajným kurzom by spôsobil vašu stratu.

Presnejšie práve preto, že nakupujete za drahší kurz než predávate späť. Menej nákladnou položkou je takzvaný „swap“, ktorý možno chápať ako poplatok, za udržiavanie obchodu, môže byť denný alebo týždenný, zväčša nemá veľký vplyv na výsledok obchodu. Táto práca sa v zásade nevenuje tomu ako tieto položky fungujú na pozadí alebo prečo vôbec existujú. Vníma ich skôr ako entity, ktoré nám ovplyvňujú obchodovanie. Pre naznačenie, swap možno chápať tiež ako úrok z držania peňazí, a to takým princípom, že ak nakúpime na príklad Austrálsky dolár na páre s Japonským yenom, tak sa stávate držiteľom Dolárov, a na základe úrokovej sadzby Austrálskej centrálnej banky, kde pokiaľ je výraznejšie kladná, tak je tu možnosť na „swape“ dokonca zarobiť. Naopak, ak by sme predávali dolár a kupovali yen, tak Japonská centrálna banka má výrazne nižšie úrokové sadzby, a teda by sme swap platili. Swap z pomedzi poplatkov, pri malých objemoch, nie je najdôležitejšou položkou, ale je vhodné si ju všimnúť.

Ďalším nastavením ktoré sa odporúča vždy využiť je zelene vyznačený „Take profit“ a „Stop loss“. Nimi si predom nastavíte ciele vášho obchodu, a bez toho aby sme museli osobne sledovať hodnotu kurzu, či sa náhodou nedotýka našich cieľových úrovní, vám visí u brokera príkaz k zatvoreniu obchodu na istej úrovni zisku alebo straty. Po rozhodnutí sa, či chcete nakupovať alebo predávať, si volíte červené alebo zelené tlačidlo a náš príkaz je odoslaný brokerovi k exekúcii. V niektorých prípadoch vstupuje do hry aj položka menom komisia (tiež jednorazový poplatok), najčastejšie je to pri nákupe akcii bez pákového efektu, zvykne sa pohybovať v rozmedzí dvoch percent ceny danej akcie.

(11)

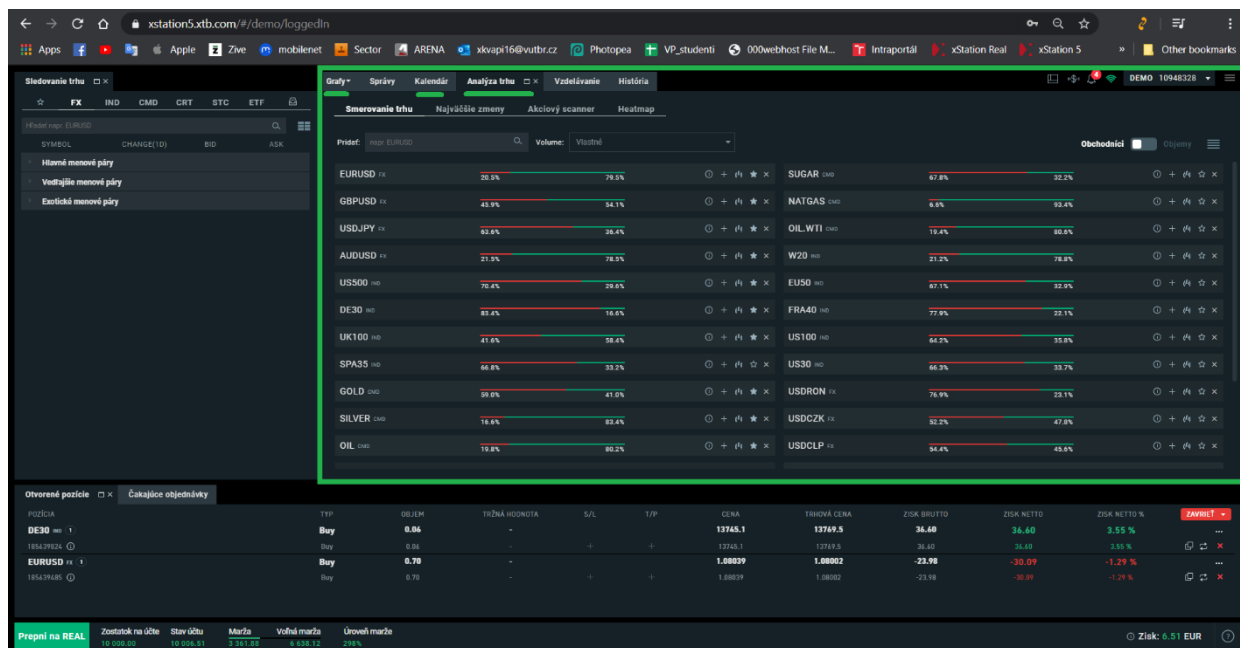
Brokeri taktiež zvyknú ponúkať správy a oznámenia o dôležitých udalostiach priamo v obchodných platformách,. Nie je výnimkou ani xStation, kde môžete vidieť mnoho udalostí a zverejňovaní výsledkov (ako napríklad z trhu práce,..), ktoré často po zverejnení majú dopad na krátkodobý vývoj kurzov. Samozrejme rozsah dopadu sa odvíja od zverejnených výsledkov. Na obrázku číslo 8 možno vidieť ako vyzerá takýto panel správ v platforme xStation 5.



Obrázok č. 8: Sledovanie aktualít v platforme

(Zdroj: Vlastné spracovanie)

Na ďalšom obrázku môžeme vidieť okno sentimentu, kde možno pozorovať náladu obchodníkov voči nákupu alebo predaju mnohých inštrumentov, to pre skúseného alebo aj pre nováčika môže byť cenným nástrojom pri rozhodovaní, pretože ísť proti trendu sa nie vždy vypláca.



Obrázok č. 9: Sentiment trhu

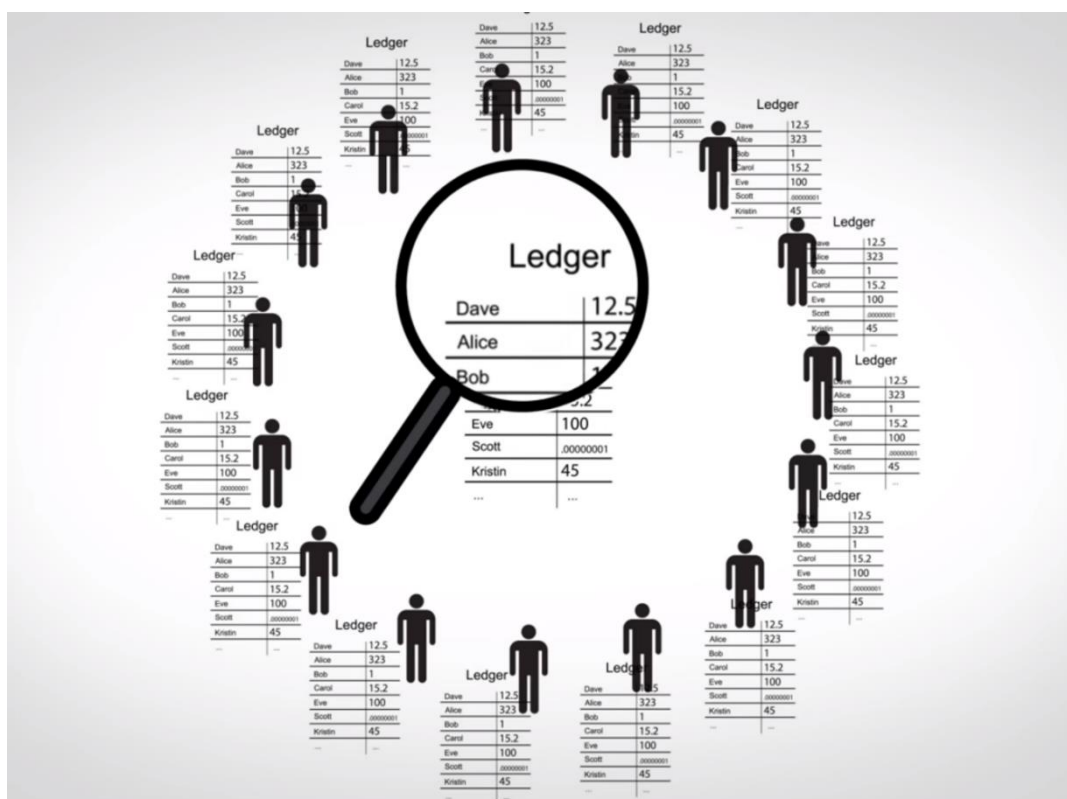
(Zdroj: Vlastné spracovanie)

1.6 Čo sú to kryptomeny

Kryptomeny alebo takzvané „Altcoiny“ sú skupinou nových transakčných systémov založených na decentralizovaných distribuovaných sieťach (teda aspoň väčšina). Decentralizovanosť značí absenciu centrálnej autority, ktorá v prípade menovej politiky zabezpečuje reguláciu objemov meny v obehu, ale aj potvrdzovanie transakcii, čo možno vnímať ako čiastočnú stratu súkromia pre jej držiteľov. Centralizácia reflektuje aktuálny bankový systém kde banka (prípadne štát) rozhoduje o tom či a ako môžeme narábať so svojimi peniazmi. Druhou vlastnosťou je distribuovanosť, možno tvrdiť, že bez nej by decentralizovanosť nebola možná, distribuovanou sieťou sa nazýva taká infraštruktúra kde každý účastník (držiteľ tokenov danej meny) vlastní kópiu informácii o zbytku siete (tzv. blockchain alebo ledger), to zabezpečuje nemožnosť podvádzať, a to už z dvoch princípov. Prvým je kryptografická podstata na ktorej je postavená kryptomena, pokiaľ by niekto chcel zmeniť bilanciu na účtoch, musel by poznať takzvané „privátne kľúče“ iných ľudí, ktoré však nepozná (pokiaľ im ich fyzicky neodcudzí), pomocou nich vieme generovať falošné transakcie ktoré by sieť akceptovala ako pravé. Druhým princípom ktorý chráni distribuovanú sieť pred podvodom je práve to, že každý drží aktuálnu kópiu celej histórie siete, a na jej zmenu by sme museli každého na sieti presvedčiť že tá naša verzia je tá pravá (čo prakticky nejde pri jej rozsiahlosti). V praxi sa jednotliví účastníci nazývajú „Node“, tí držia kópiu blockchainu a taktiež svoje vlastné kľúče, na sieti načúvajú „Minerom“ ktorí rozposielajú aktualizácie obsahu blockchainu vo forme nových blokov (skupiny nových transakcii), samotné „Nodes“ si vždy jednoducho overia kryptografickú správnosť nového bloku a zaradia ho do svojho blockchainu. V zásade platí že najdlhší blockchain je tým správnym, preto že má za svojím vybudovaním najviac práce a teda ak sa vygenerujú dva rôzne a však validne bloky naraz, majme bloky A a B obsahujú rôzne transakcie, avšak oba sú platné, naslúchajúce „Nodes“ prijímajú ten blok ktorý k nim dorazí prvý, stane sa však to, že niektoré „Nodes“ dôverujú bloku A a iné zas B, vtedy sa teda čaká na ďalší blok ktorý má štatisticky veľmi malú pravdepodobnosť že by vyšiel naraz s iným, nasledovný blok by teda bol nasledovníkom jednej z dvoch verzii blockchainu, tá s ktorou by bol spätý sa stáva dlhšou a teda ostatné „Nodes“, ktoré ju predtým neuznávali sa k nej automaticky pridávajú. Pri vytváraní bloku „mineri“ plnia blok transakciami (viac-menej náhodne), potom k bloku generujú (ťažia) špecifický „seed“ pre digitálny podpis, ktorý je podmienený značnou výpočtovou zložitou (5)

1.6.1 Čo je to Bitcoin

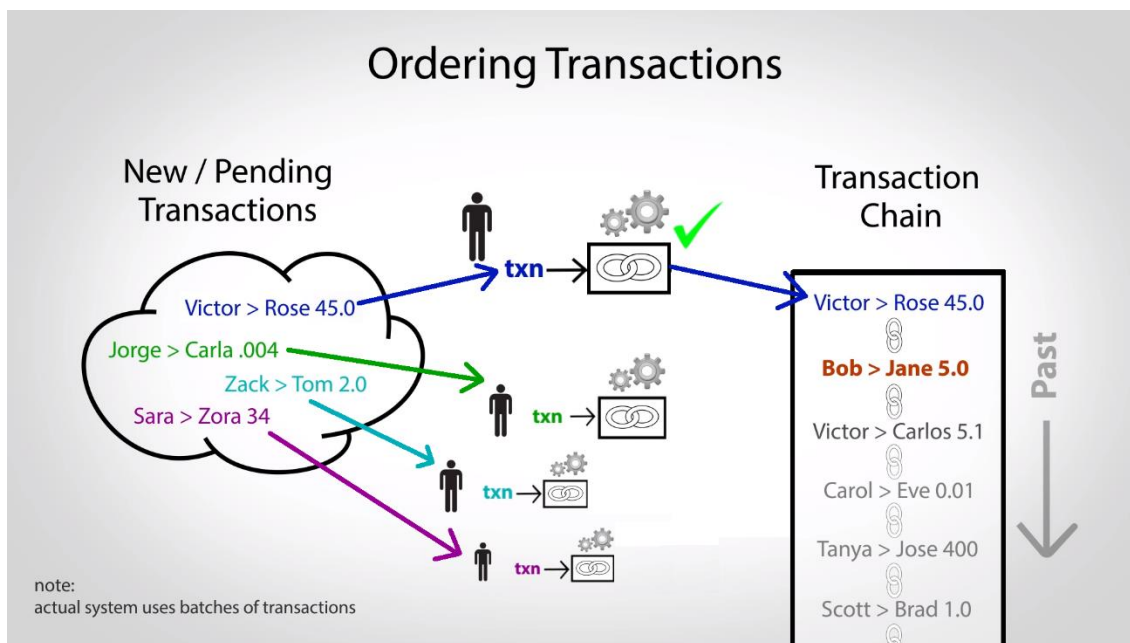
Je najznámejšou kryptomenou, má neznámeho autora, možno ju považovať aj za prvú svojho druhu, tieto vlastnosti jej dávajú výhodu oproti ostatným kryptomenám ktorých autori sú známi a teda komunita nemusí mať dôveru v integritu nimi vytvoreného systému. Bitcoin predstavuje open-source riešenie transakčnej problematiky. Splňuje vlastnosti spomenuté v predošlom odseku, tiež možno tvrdiť, že priniesol Blockchain technológiu do povedomia priemyslu a verejnosti. Fungovanie Bitcoinovej siete je pre z používateľského hľadiska veľmi jednoduché. Ak chcete držať Bitcoinové tokeny, musíte vlastniť Bitcoinovú peňaženku, táto peňaženka pozostáva z dvojice kľúčov, verejného a súkromného. Verejný možno chápať aj ako číslo účtu a privátny ako heslo k nemu. Infraštruktúra Bitcoinu závisí na tom, aby držitelia peňaženiek udržiavali kópiu takzvaného „blockchain ledgeru“, to možno chápať ako zoznam predošlých transakcií tejto siete, z neho možno vyčítať bilancie všetkých účastníkov, samozrejme skrytých pod pseudonymami (pomocou svojich verejných kľúčov). Na obrázku možno vidieť ilustráciu distribuovanosti siete, v realite sú však mená reprezentované verejnými kľúčmi. (5)



Obrázok č. 10: Ilustrácia distribuovanej siete

(Zdroj: 6)

Pri posielaní peňazí, sa táto akcia „broadcastuje“ do celej P2P siete, aby každý zistil, že posielate peniaze. Vtedy do hry vstupujú takzvaní „Mineri“, ktorí potvrdzujú tieto transakcie. Aby nebolo možné poselať peniaze iných než vaše, každý príkaz k transakcii má digitálny podpis pomocou odosielateľovho privátneho kľúča. Minerí potom pomocou svojej výpočtovej sily dokážu overiť pravosť tohto podpisu. Skupiny platných transakcií uzatvárajú do blokov, ktoré potom rozposielajú do siete. V praxi taktiež nemusíme udržiavať svoj blockchain aktuálny, sú totiž firmy, ktoré udržiavajú naše peňaženky v aktuálnom stave, dnes veľkosť Bitcoinového blockchainu predstavuje okolo 240gb. Pri spracúvaní transakcií sa zapisujú do vytváraných blokov práve tie transakcie ktorých digitálne podpisy sú pravé. Poradie vybraných transakcií do blokov nejde ovplyvniť, sú v zásade vyberané náhodne. Jedným trikom ako dostať svoju transakciu čo najskôr do bloku, je pridať k nej voliteľný parameter ktorým je poplatok za transakciu, tento poplatok dostane ten Minner ktorý uvedie danú transakciu do bloku a úspešne ho uzavrie včasným vyťažením a rozposlaním po sieti ostatným na kontrolu, ktorí ho skontrolujú a zaradia do svojho blockchainu (14)



Obrázok č. 11: Spracovanie transakcií „from pool to block“

(Zdroj: 6)

1.7 Ako obchodovať s kryptomenami

Pri obchodovaní s kryptomenami však nemusíte rozumieť ich úplnému pozadiu, môžete ich nakupovať tak, že ich nebudete fyzicky držať kľúče od peňaženiek ale vám ich bude spravovať spoločnosť ktorá prevádzkuje burzu. Pri predávaní si ich spätne rozmeníte na fiat menu. Možno tak urobiť cez mnohé firmy, ktoré ponúkajú správu peňaženiek, nie je to však najspoľahlivejším riešením. Burzy sú výhodným cieľom pre hekerov, a teda po nákupe je najvhodnejšie si svoje tokeny poslať na vlastnú peňaženku. Mňa v tejto práci však bude zaujímať dostupnosť obchodovania s kryptomenami na burze. Momentálne je prístup ku kryptomenám veľmi jednoduchý a najznámejšie kryptomeny nájdeme na burzách často aj medzi menovými pármí. Možno s nimi narábať úplne rovnako ako akýmkoľvek inými inštrumentami.



Obrázok č. 12: Bitcoin na burze

(Zdroj: Vlastné spracovanie)

1.8 Čo sú to technické indikátory

Informácie ktoré nám poskytujú indikátory sú výstupmi matematických funkcií, ktoré analyzujú zväčša voľne dostupné cenové a objemové dáta daného aktíva, ich účelom je poskytnúť nám kontinuálne prepočty reflektujúce aktuálny stav trhu, často sú vizuálne reprezentované v grafe.

1.8.1 Jednoduchý kľzavý priemer SMA

Technické indikátory môžeme rozdeliť do niekoľkých základných skupín, najpoužívanejšou skupinou sú takzvané kľzavé priemery („Moving Average“) u ktorých samotný názov môže indikovať princíp fungovania. Najjednoduchším príkladom je Jednoduchý kľzavý priemer („Simple Moving Average“) ktorý v princípe počíta priemernú cenu za niekoľko dní a to tak, že vezmeme zatváracie ceny za niekoľko dní a pomocou jednoduchého aritmetického priemeru vypočítame priemernú hodnotu, tým že tento priemer vieme vypočítať pre každý deň, môže nám z týchto hodnôt vzniknúť graf ktorý zvykne mať podobu vln pričom dôležitým parametrom je práve to koľko dní spätne sledujeme alebo takzvaná perióda, v praxi sa zvykne používať 5,10,20,50,100,120 a 200 dňová perióda, samozrejme kľzavý priemer možno aplikovať aj na minútové alebo hodinové rámce ak skúmame krátkodobé trendy. Nasledujúci obrázok ukazuje príklad SMA. (2)



Obrázok č. 13: Jednoduchý Kľzavý Priemer

(Zdroj: 8)

Mnoho obchodníkov a teda značná časť trhu zvykne sledovať tento indikátor a teda si môžeme pri vhodne zvolenej perióde všimnúť že sa cena od hodnoty tohto indikátoru môže odraziť alebo nám indikovať kedy je dané aktívum prekúpené alebo podhodnotené. V štandardných prípadoch kedy trhom nehýbu dôležité udalosti ktoré neberú v ohľad na

aktuálnu situáciu možno vcelku spoľahlivo využiť tohto indikátoru alebo jeho rôznych modifikácií v navigovaní sa pri obchodovaní. (10)

Kĺzavý priemer tiež možno upraviť a to napríklad do podoby Váženého kĺzavého priemeru kde pri výpočte dáme starším dátam menšiu váhu. (2)

1.8.2 Bollingerové pásmo

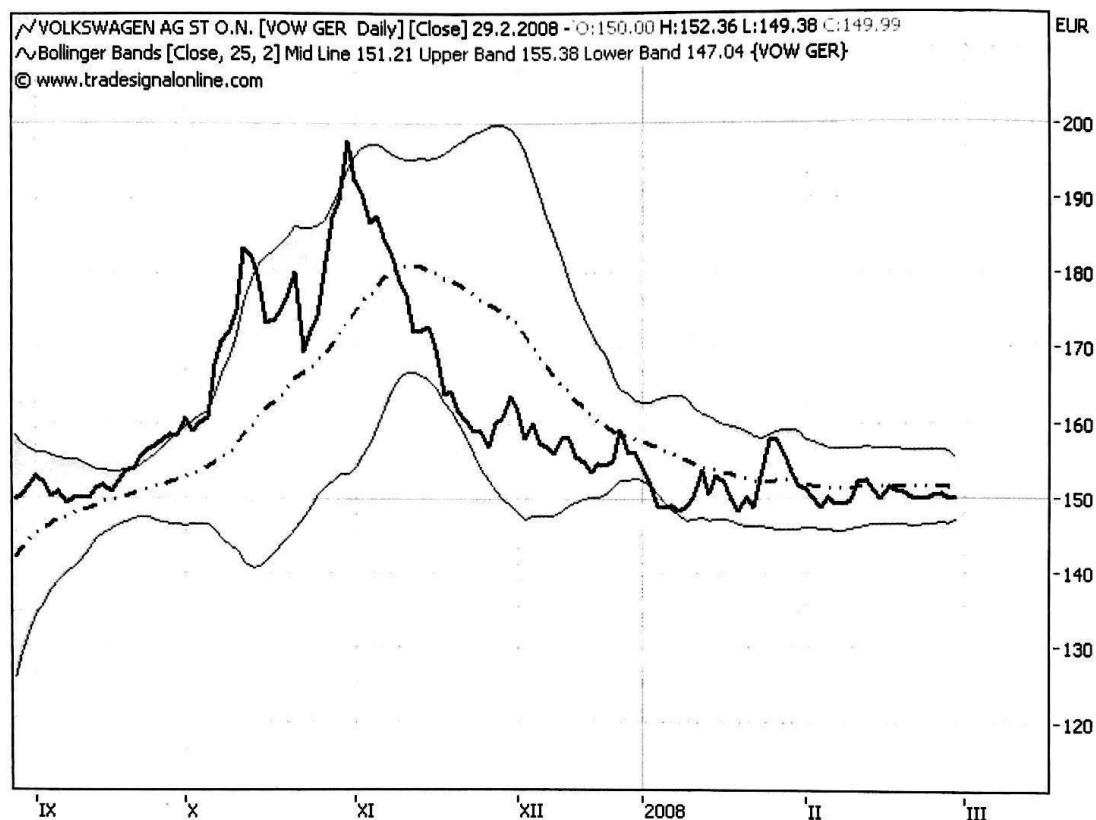
Bollingerove pásmo („Bollinger band“) je pásmový indikátor, u ktorého sa šírka obálky mení v závislosti na volatilita kurzu predmetnej akcie. Napriek tomu, že umožňuje používať rôzne druhy kĺzavých priemerov, v praxi sa takmer vždy výlučne odvodzuje od jednoduchého kĺzavého priemeru. (2, s. 335)

Praktické využitie Bollingerovho pásma ukazuje nasledujúci obrázok č. 13, jedná sa o vývoj akcií spoločnosti Volkswagen, pričom základom pásma je 25-dňový jednoduchý kĺzavý priemer a hodnota smerodatnej odchylky je 2. (2, s. 336)

Uvedený graf z nasledujúceho obrázku možno interpretovať takto:

1. Medzera medzi jednotlivými pásmami je založená na volatilita kurzu. V prípade, že sa priestor medzi pásmami zúži, je táto fáza označovaná ako konsolidačná, po nej sa v blízkej dobe očakáva prudká zmena kurzu. V uvedenom prípade dochádza k zreteľnému stiahnutiu pásma iba v mieste označenom hviezdíčkou. V tomto prípade je signál potvrdený.
2. Ďalší význam tohto indikátoru spočíva v generovaní nákupných a predajných signálov v tých prípadoch, kedy kurz prekročí hranice pásma a následne sa vráti späť. V danom prípade sú generované 3 predajné signály a jeden nákupný. Pritom je zrejmé, že prvý predajný signál je falošný.

3. Tretím sledovaným faktorom je približovanie sa kurzu k hranici pásma. Aj na základe tohto faktoru je možné generovať nákupné a predajné signály, má sa však všeobecne za to, že nebývajú príliš spoľahlivé (čo tiež uvedený graf potvrdzuje). (2, s. 336)



Obrázok č. 14: Bollingerové pásmo

(Zdroj: 2, s. 336)

1.8.3 Oscilátory

Oscilátory sú technickými indikátormi merajúcimi zmenu ceny (kurzu) za zvolené časové obdobie. Pre ich výpočet je samozrejme potrebné neprerušovanej a dostatočne dlhej časovej rady vstupných dát, pričom problémom býva aj správna voľba jej hornej a dolnej hranice (ktorej prekročenie býva signálom k nákupu alebo predaju predmetných akcií) (2)

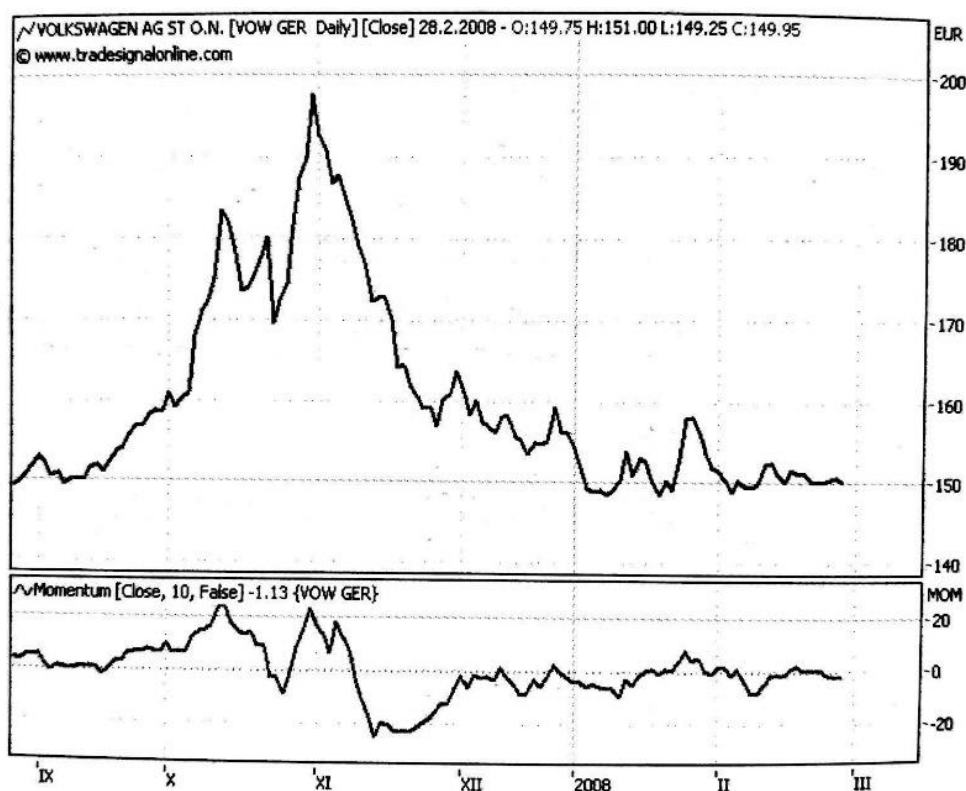
Napriek tomu, že oscilátory možno používať aj na býčích alebo medvedích trhoch (kde sa však často indukujú falošné signály), sú vhodné predovšetkým na postranné trhy (kde naopak mnoho iných technických prostriedkov zlyháva). A ďalšou ich významnou

spoločnou vlastnosťou je (rovnako ako u kľzavých priemerov), že čím kratšia je dĺžka zvoleného obdobia, tým viac signálov indukujú (a naopak). (2)

Oscilátorov existuje značné množstvo, medzi najznámejšie a v praxi najpoužívanéjšie patria predovšetkým:

- Momentum
- Rate of Change (ROC)
- Relative Strength Index (RSI)
- Moving Average Convergence Divergence (MACD)

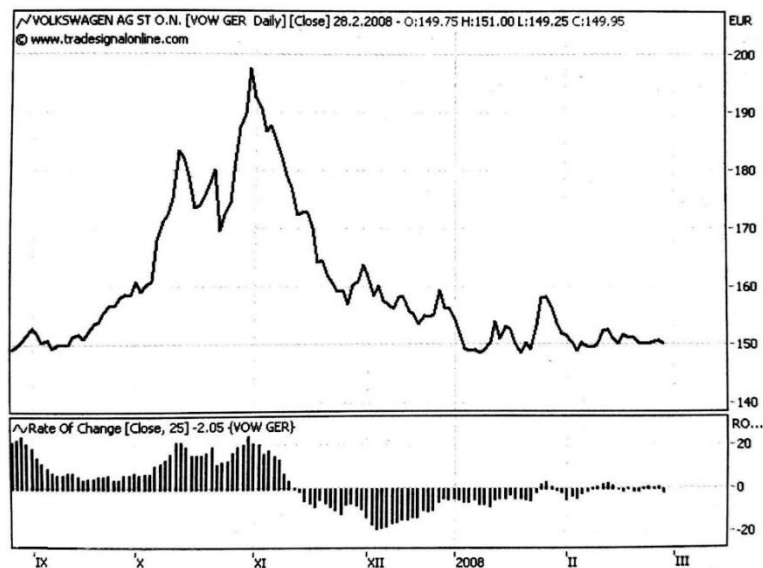
Momentum je najjednoduchším oscilátorom, ktorý meria zrýchlenie alebo spomalenie trendu na základe vzájomného porovnania aktuálnych zatváracích kurzov akcií s kurzami platnými na začiatku sledovaného obdobia. (2, s. 337)



Obrázok č. 15 Momentum Oscilátor

(Zdroj: 2, s. 338)

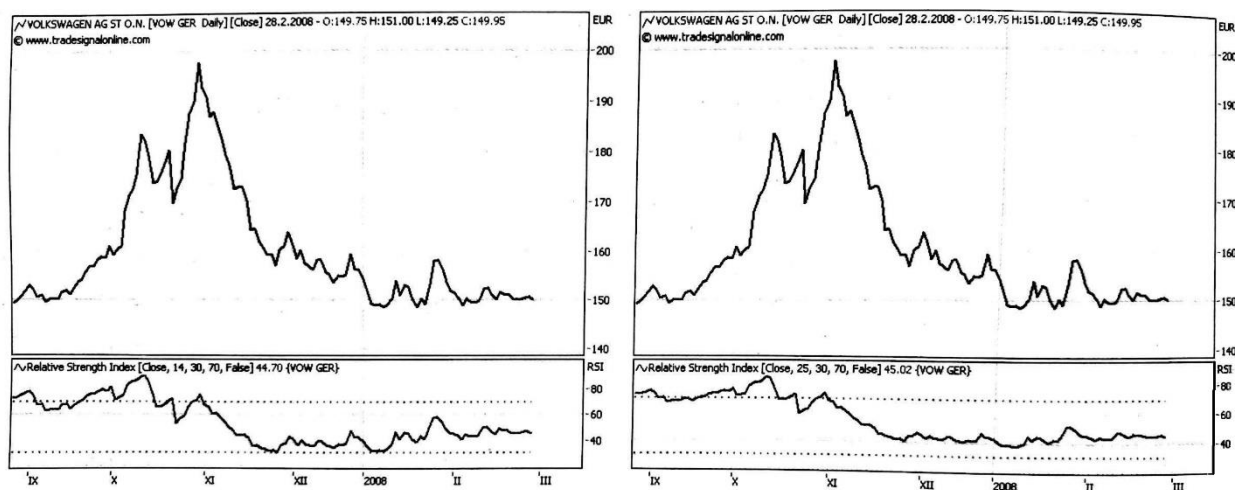
Rate of Change oscilátor je veľmi podobný oscilátoru typu Momentum pričom udáva relatívnu zmenu súčasného akciového kurzu vzhľadom na hodnotu kurzu platného pred zvoleným počtom obchodných dní. (2, s. 338) Môžete si ho všimnúť na obrázku nižšie.



Obrázok č. 16 Rate of Change Oscilátor

(Zdroj: 2, s. 339)

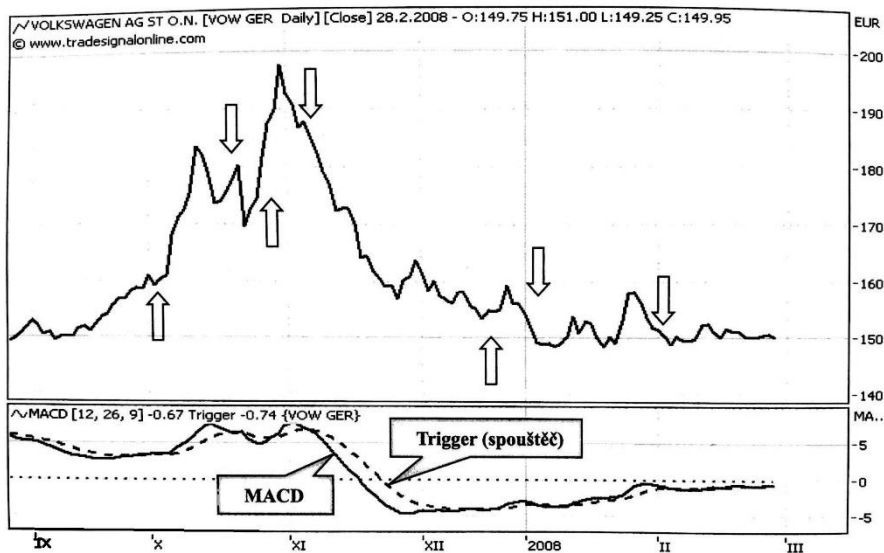
Relative Strength Index alebo aj „Index relatívnej sily“ je pomerne jednoduchý z hľadiska výpočtu ale náročný na interpretáciu. Bol zostavený za účelom odstránenia obecných známych nedostatkov oscilátorov a slúži k identifikácii trendov a k indikácii signálov odporúčajúcich vstup na trh, respektíve jeho opustenie. (2, s. 339)



Obrázok č. 17 Relative strength Index Oscilator

(Zdroj: 2, s. 340)

Moving Average Convergence Divergence je obecné považovaný za jeden z najspôhlivejších indikátorov technickej analýzy. Vytvára sa odpočítaním hodnoty dlhodobého exponenciálneho klzavého priemeru (z pravidla 25 alebo 26 denného) od hodnoty od hodnoty obdobného, však krátkodobého, idikátoru (obvykle 12 dňového). (2, s. 340)



Obrázok č. 18 MACD Oscilátor

(Zdroj: 2, s. 342)

1.9 Analýzy

V tejto časti predstavím podstatné okruhy a entity súvisiace s analýzami a analytickými postupmi.

1.9.1 Psychologická analýza

Psychologická analýza sa zaoberá mentálnym stavom a emóciami obchodníka, psychika je dôležitým faktorom ktorý ovplyvňuje úspech počas obchodovania. Psychická stabilita počas obchodovania môže byť rovnako dôležitá, ako vedomosti a skúsenosti. Disciplína a schopnosť zniesť riziko, sú dvoma najdôležitejšími aspektami tejto analýzy. Zatiaľ čo strach a slepé nadšenie z vidiny zisku, sú dvomi najznámejšími emóciami spojenými so psychológiou pri obchodovaní, ďalšími emóciami ovplyvňujúcimi správanie sú nádej a sklamanie. (1)

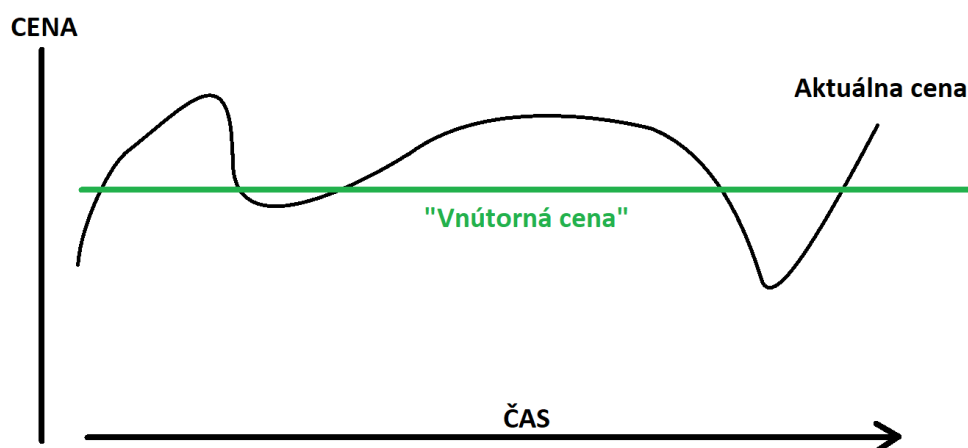
Vidina zisku vyvoláva nezastaviteľnú túžbu po bohatstve. Pri snahe dosiahnuť čo najvyšších ziskov, vďaka tejto emócii, obchodníci zostávajú v otvorených obchodoch dlhšie než plánovali. Túto ľudskú vlastnosť možno pozorovať počas posledných fáz býčich trendov. (1)

Strach je emóciou tak silnou, že dokáže obchodníkov zviest' zo svojej naplánovanej stratégie a vedie obchodníkov k predčasnému uzatváraniu pozícií z obáv veľkých strát. Strach na trhu je možné pozorovať počas medvedieho trendu kde vedie obchodníkov k iracionálnym aktom v snahe vystúpiť z obchodu. (1)

Sklamanie, či už z premeškaných príležitostí alebo stratových obchodov, môže skresliť obchodníkov úsudok a dohnáť ho k vstupu do obchodu po rýchlom raste ceny. Toto konanie vedie väčšinou ku rovnako rýchlym stratám. (1)

1.9.2 Význam fundamentálnej analýzy

Fundamentálnu analýzu mnohí považujú za najkomplexnejšiu analýzu ktorú obchodník môže urobiť, používa sa najmä pri prijímaní zásadných investičných rozhodnutí. Je založená na predpoklade, že vnútorná (teoretická) hodnota daného aktíva je iná než je jej aktuálna cena za ktorú sa obchoduje na finančných trhoch. Pre pochopenie pojmu „vnútorná hodnota akcie“ a pochopeniu podstaty fundamentálnej analýzy, je nutné si uvedomiť že sa jedná len o individuálny názor účastníka akciového trhu na to aký by mal byť takzvaný „spravodlivý kurz“, ktorého hodnotu v krátkom časovom rámci možno považovať za nemennú v porovnaní s meniacim sa tržným kurzom (2)



Obrázok č. 19: Znázornenie vzťahu vnútornej a reálnej ceny

(Zdroj: Vlastné spracovanie podľa: 1, s. 220)

To znamená že pokiaľ podľa fundamentálnej analýzy zistíte, že vnútorná hodnota daného aktíva je vyššia než je jeho tržná cena, možno dané aktívum považovať za podhodnotené a teda uvažovať nad jeho nákupom. Na druhú stranu ak podľa analýzy zistíte, že je dané aktívum predražené, teda nadhodnotené, možno ho, buď predat', alebo ak máte v pláne len nakupovať, tak sa od neho chvíľu držať ďalej. (2)

Priebeh vytvárania fundamentálnej analýzy nie je rovnaký u každého aktíva, pri vytváraní fundamentálnej analýzy sa berú do úvahy rôzne skutočnosti ohľadom sledovaného aktíva, počínajúc správami v mediach, až po analýzu finančného zdravia firmy emitujúcej akcie. Tým pádom platí, že každý obchodník môže vyhodnotiť vnútornú hodnotu daného aktíva inak, a to aj v prípade, keď používajú podobné postupy, pretože jednotlivým informáciám a výsledkom môžu dávať subjektívnu váhu. Táto nepresnosť teda spôsobuje nestálosť ceny daného aktíva, a teda že každý obchodník nakupuje na iných cenových hladinách a očakáva iné ciele. (2)

1.9.3 Význam technickej analýzy

Technická analýza v zásade vychádza z publikovaných verejných údajov o danom aktíve, najčastejšie je to vývoj ceny a objemy za ktoré sa obchodovalo, prípadne ďalšie parametre týkajúce sa vybraného aktíva. (11)

Technický analytici vychádzajú z predpokladu, že správanie obchodníkov ktorí obchodujú na základe technických analýz je veľmi podobné a často sa zvykne opakovať. Pri analyzovaní ceny aktíva si možno všimnúť, že v grafe sa často opakujú rôzne javy ktoré zvykneme nazývať „patterny“ (vzory), na základe týchto vzorov zvyknú obchodníci vynášať investičné rozhodnutia a predikcie budúcej ceny. Na rozdiel od fundamentálnej analýzy obchodníci veria tomu že vývoj ceny aktíva je možné predpovedať na základe podobnej mentality ľudí a predošlého správania sa ponuky a dopytu. Všeobecne platí, že ľudia zvyknú fundamentálnu analýzu považovať za príliš zložitú alebo príliš subjektívnu nato aby sme vedeli pravidelne predpovedať budúcu cenu a teda zakladajú práve na technickej analýze, ktorá zásadne vychádza z historického vývoja ceny. (9)

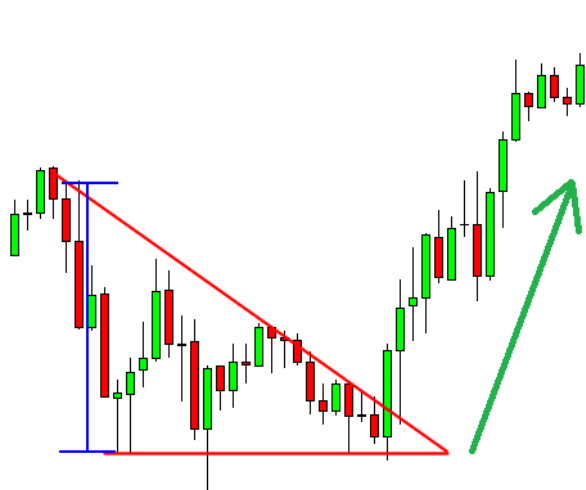
Pri obchodovaní na základe technickej analýzy sa zvykne rozpoznávať niekoľko všeobecne známych vzorov (patternov), ako je možné vidieť na obrázkoch nižšie. Ak ale máte pozorné oko a dostatok paranoje môžete objaviť aj nejaké vlastné. Na obrázku môžete pozorovať pattern Head and Shoulders, ktorý zvykne indikovať predajný signál.



Obrázok č. 20: Head and Shoulders pattern

(Zdroj: Vlastné spracovanie podľa 13)

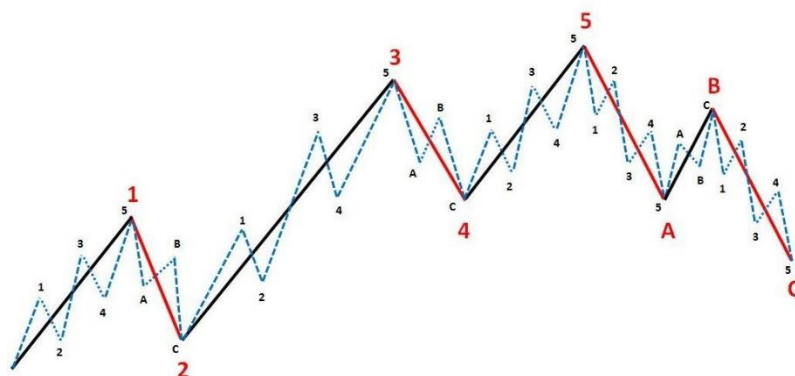
Ďalší obrázok nám ukazuje pattern nazývaný Triangle, ktorý samozrejme disponuje aj inverznou variantou ako aj podobne tvarovanými variantami indikujúcimi opačné signály. Treba však mať na pamäti, že výskyty patternov sa líšia inštrument od inštrumentu.



Obrázok č. 21: Triangle pattern

(Zdroj: Vlastné spracovanie)

Významným vzorom u technickej analýzy sú takzvané „Elliottove vlny“ ktoré popisujú pohyb kurzu vo veľmi špecifických vzoroch spôsobených pohybom dopytu a ponuky v pravidelných cykloch, v praxi to znamená, že ak máme napríklad rastúci trend tak nárast ceny prebieha s pravidelnými poklesmi, ktoré možno interpretovať ako prekúpenie daného aktíva. Vtedy cena na chvíľu klesne na úroveň, kde začne byť braná opäť ako podhodnotená, a vtedy obchodníci začnú znova nakupovať. Teória Elliottových vln je však omnoho zložitejšia než tento jednoduchý popis a zabrala by kapitolu sama o sebe, pre účely tejto práce ju spomínam ako rozšírenie povedomia o analytických technikách. Na obrázku nižšie, si možno všimnúť pohyb ceny, pozostávajúci z pravidelných nárastov a poklesov ktoré sú práve nazývané „Elliottove vlny“. (7)



Obrázok č. 22: Elliottove vlny

(Zdroj: 7)

Výhodou technickej analýzy je teda práve to, že ju používajú mnohí obchodníci, teda tým, že mnoho ľudí obchoduje na základe rovnakých pravidiel a techník, zvyknú vynášať podobné rozhodnutia. To v praxi znamená, že ak všetci nakupujú, tak cena skutočne môže stúpať vyššie a teda stratégia plní účel už len z toho princípu, že obchodníci výrazne zvýšia dopyt po danom aktíve.

Nevýhodou technickej analýzy však môže byť neschopnosť predikovať výrazné prepady ktoré môžu spôsobiť napríklad politické konflikty a iné nepredvídateľné situácie čo v praxi znamená, že napríklad pri predikcii ceny ropy nevieme odhadnúť či zajtra niekto neodpáli ropné pole na blízkom východe, samozrejme pri fundamentálnej analýze to nezistíme tiež ale berieme do úvahy práve geopolitické dianie ktoré nám môže naznačiť vysokú volatilitu a teda sa obchodu vyhnúť.

2 ANALÝZA SÚČASNÉHO STAVU

Z tejto časti je možné lepšie porozumieť dôvodu vytvorenia tejto práce na zvolenú tému. Tiež predstavím, mne známe dostupné možnosti pre algoritmizáciu technickej analýzy a rozpoznávania paternov.

Aktuálne, máme štandardne k dispozícii u väčšiny brokerov možnosť skúmať cenu pomocou čiarových alebo sviečkových grafov, máme k dispozícii desiatky brokerov kde každý z nich využíva buď vlastnú webovú aplikáciu alebo inštalovateľnú v počítači (štandardne MetaTrader4). Ako doplnok už dnes skoro každý broker poskytuje mobilnú aplikáciu aby mohol byť obchodník vždy v strehu.

Pri vynášaní rozhodnutí obchodník zvykne využívať rôzne analýzy ktoré by ho mali naviesť k správne rozhodnutiu. Zvolené analýzy vychádzajú práve z obchodníckej stratégie a ich rozsah je obmedzený iba jeho schopnosťami.

Obchodník často narazí na situáciu kde je analýza príliš namáhavá alebo časovo náročná na ručné spracovanie, najmä keď pre daný prípad neexistuje žiaden voľne dostupný nástroj ktorý by mu s tým pomohol.

Častou slabou stránkou obchodníka je práve počítačová zručnosť, a to najmä v oblasti programovania, tým pádom jeho vlastné stratégie sú obmedzené obchodníkovi vlastnými časovými možnosťami kedy musí sám fyzicky sledovať vývoj ceny a včas reagovať na zmeny ceny. U menej zložitých stratégií nie je problém sledovať cenu pomocou jednoducho nastaviteľných notifikácií, kde vám telefón pomocou notifikácie oznámi prípad, keď sa cena priblíži sledovanej hodnote, taktiež obchodník môže využívať obchodné príkazy typu limit a stop, kde sa nákup a predaj vykonajú splnením jednoduchých podmienky.

Ak však obchodníckova stratégia vyžaduje príliš častú pozornosť, napríklad na báze každej hodiny alebo niekoľko-minútových intervalov, stávajú sa tieto postupy nepraktickými a hlavne nespoľahlivými. Vždy sa môže stať, že obchodník premešká príležitosť ak má v tom čase niečo iné na práci.

Pri manuálnom obchodovaní je taktiež veľmi časovo náročné sledovať zložitejšie úkazy. To sú napríklad opakujúce sa vzory v grafoch, vtedy je možné na základe týchto vzorov obchodovať (pokiaľ sa dostatočne často opakujú).

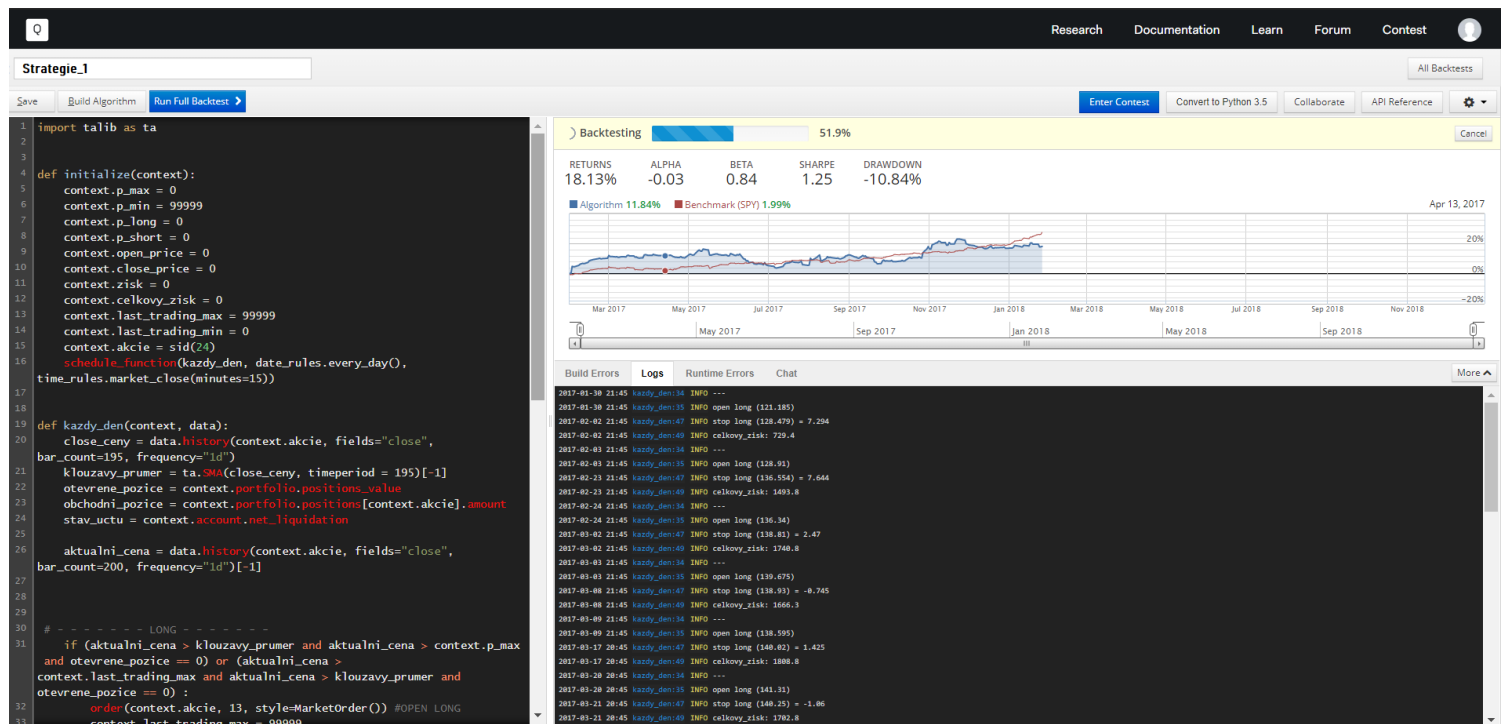
U rôznych inštrumentov sa dajú pozorovať rôzne vzory, taktiež sa tieto vzory môžu nachádzať na rôznych škálach, od minútových časových rámcov (time frame) až po týždenné, teda je na obchodníkovi či jeho stratégia zakladá na dlhodobom obchodovaní alebo na dennej báze, podľa toho si vyberá škálu (time frame) na akej sleduje cenu.

2.1 Dostupné platformy pre algorimické obchodovanie

Z bežných dostupných platforiem, či už webových alebo inštalovateľných, v základe žiadna neposkytuje možnosť sledovania vzorov (patternov) ani na statickej báze (špecifická postupnosť sviečok danej veľkosti) a už vôbec nie na dynamickej báze (kde sviečky formujú istý pattern). Pokiaľ ale obchodník disponuje istou znalosťou programovania, je tu mnoho nástrojov v ktorých sa môže realizovať, za zmienku stoja nasledovné:

2.1.1 Quantopian

Quantopian ktorý poskytuje webové rozhranie a vlastné API na báze programovacie jazyka Python a tiež s možnosťou využitia nimi sprostredkovaného Jupyter Notebooku, táto platforma má však aj svoje nevýhody ako napríklad obmedzenia niektorých príkazov hlavne pre to aby sa od nich nedali kradnúť dáta. Ďalším obmedzením je ich biznis model. Dovolia vám totiž používať svoje prostredie na testovanie stratégií, avšak vám nedovolia investovať vlastné peniaze do vašej stratégie. Možno je ale, sa prihlásiť do súťaže, kde potom oni sami môžu zafinancovať vašu stratégiu a následne máte nárok na časť profitu. Obmedzenia z pohľadu užívateľskej prívetivosti by som na platforme tiež našiel. Vytkol by som to, že pri testovaní nemáte vizualizáciu obchodov v cenovom grafe a teda neviete, či sa obchody vykonávajú len vtedy, keď by mali. Máte dostupnú iba krivku profitu ako výstup z vašej stratégie a možnosť exportu pokročilého ekonomického vyhodnotenie ako následný report kde zistíte štatistické údaje o vykonaných obchodoch vo veľmi reprezentatívnej forme, na nasledujúcom obrázku uvidíte vývojové prostredie quantopianu.



Obrázok č. 23: Quantopian

(Zdroj: Vlastné spracovanie)

2.1.2 Quantconnect

Ďalšou platformou je Quantconnect. V ňom sa podobne ako v Quantopiane programuje v prevažne Pythone. Majú však značne iné API než Quantopian, tak že nie je zárukou to, že ak ovládáte jedno, tak vám ľahko pôjde aj druhé. Poskytujú rovnako možnosť práce s Jupyter Notebookom, zdieľanie projektov a pokročilé štatistické reporty. Biznis model Quantconnectu je však otvorený živému obchodovaniu a platforma tiež poskytuje trochu lepší náhľad do equity grafu. Stále ju nehodnotím ako úplne najschodnejšiu cestu, a to preto, stále chýbajúca vizualizácia obchodov a API a UI je výrazne orientovaná na skúsenejších programátorov, miesto toho aby bolo postavené ako doplnujúci prvok k obchodovaniu, tiež má relatívne malú komunitu a teda pre prípadný debugging kódu, to nie je najlepšie prostredie. V obrázku nižšie môžeme vidieť ukážku z prostredia Quantconnectu.



Obrázok č. 24: Quantconnect backtest výsledok

(Zdroj: Vlastné spracovanie)

2.1.3 MetaTrader 4

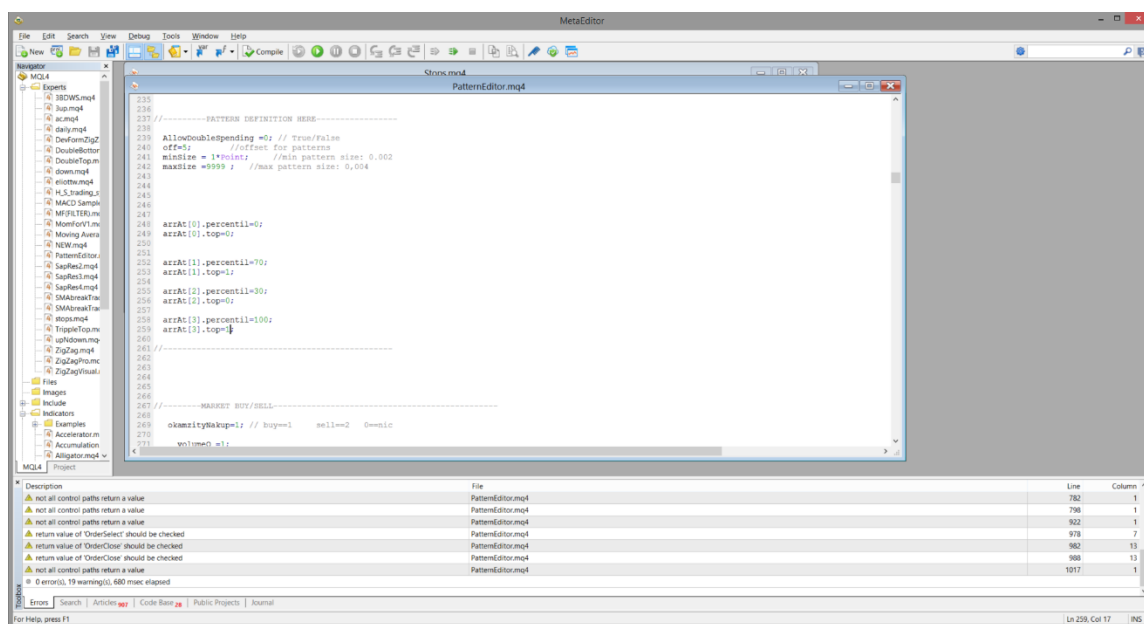
Platformou, na ktorej som sa rozhodol vytvoriť svoje riešenie, je teda známy MetaTrader4. Ten je nutné nainštalovať na počítač. Jeho výhodou je, že takmer každý štandardný broker poskytuje možnosť spojenia s touto platformou. Ďalšou výhodou je to, že je silne orientovaný na funkčnosť, užívateľ má možnosť nastaviť si vizuál grafov, vkladať doň čiary a rôzne iné objekty spojené s technickou analýzou, taktiež je možné si do tejto platformy stiahnuť indikátory tretích strán alebo si vlastné naprogramovať. Najzaujímavejšou funkciou pre náročného užívateľa je však takzvaný expertný poradca (Expert Advisor), ktorý dovoľuje naprogramovať si vlastnú stratégiu v jazyku MQL4, ktorý je založený na C# syntaxi. Stratégiu potom môžete dovoliť živé obchodovanie cez váš účet. Má aj možnosťou testovania (backtest) si stratégie na historických dátach predtým než ju pustíte na ostro, taktiež máte možnosť optimalizovať parametre vo vašej stratégii, a to vo veľmi priateľskom prostredí. Teda nemusíte spustiť backtest, skontrolovať výsledky, následne upraviť parametre a spustiť test znova. Stačí si predvoliť hodnoty a MetaTrader ich sám otestuje, vy si už len skontrolujete a vyberiete najlepšie výsledky. V ďalšom obrázku môžete vidieť obchodné rozhranie Metatraderu 4.



Obrázok č. 25: MetaTrader 4

(Zdroj: Vlastné spracovanie)

MetaTrader ponúka v samostatnej aplikácii editor kódu, menom MetaEditor. Z ďalšieho obrázku môžete pozorovať, že sa vizuálne jedná o štandardný editor kde je možné písať kód, taktiež kompilovať a zároveň cez MetaEditor aj exekuovať kód. Funguje to tak, že si MetaTrader a MetaEditor spustíme naraz, pri exekúcii kódu MetaEditor pošle tento kód MetaTraderu a v ňom možno sledovať priebeh programu.



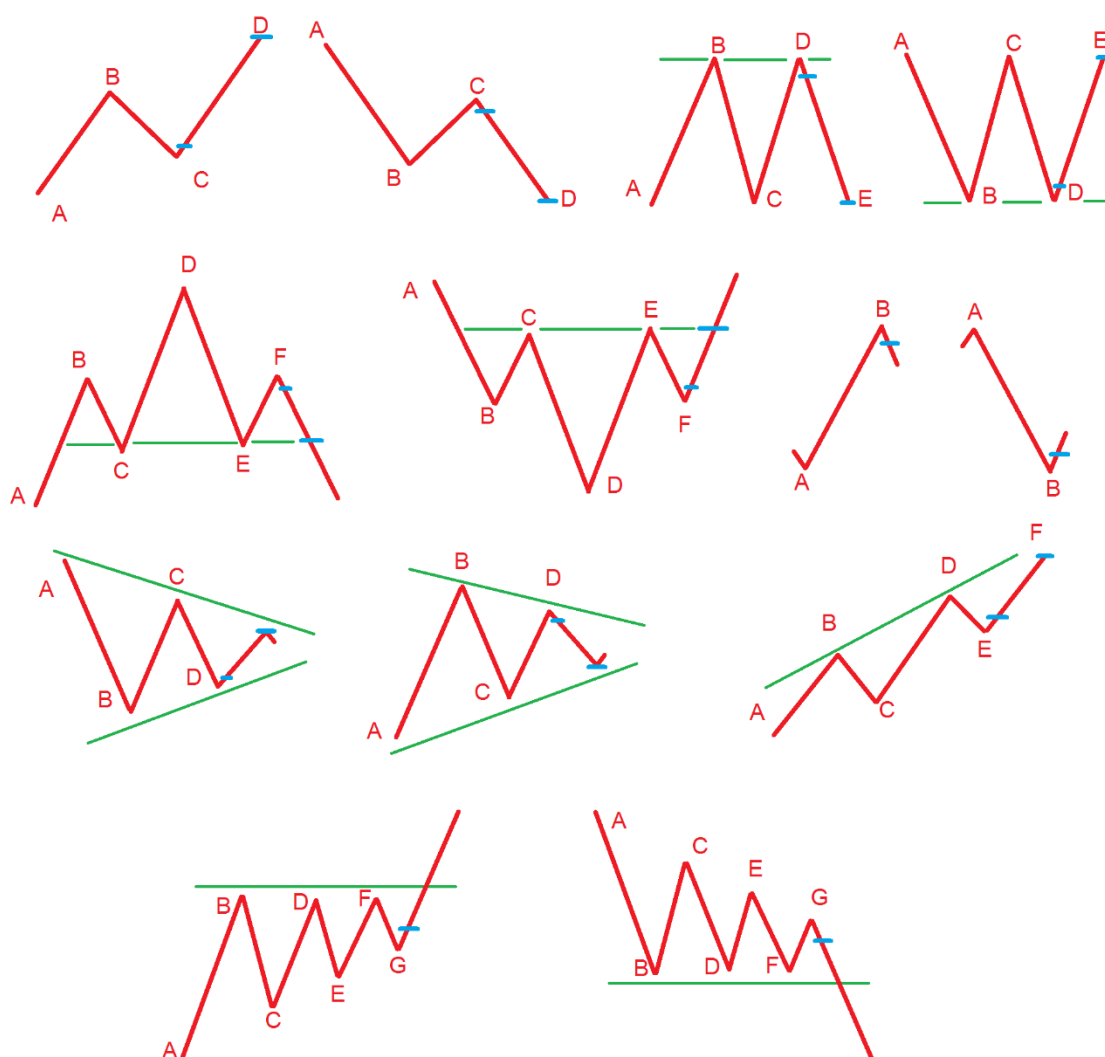
Obrázok č. 26: MetaEditor

(Zdroj: Vlastné spracovanie)

3 VLASTNÉ NÁVRHY RIEŠENIA

V tejto časti sa pokúsim podrobne vysvetliť návrh a implementáciu svojho riešenia k problematike automatického obchodovania na technických analýzach.

Výstupom tejto práce je program ktorý dovoľuje užívateľovi definovať si vzor, ktorý by rád monitoroval a prednastaviť akciu ktorá sa po jeho objavení automaticky spustí v reálnom čase. U jednotlivých vzorov bude možné definovať relatívnu výšku jednotlivých uzlov a toleranciu pri ich rozpoznávaní. Príklady niektorých možných vzorov možno vidieť v obrázku nižšie.

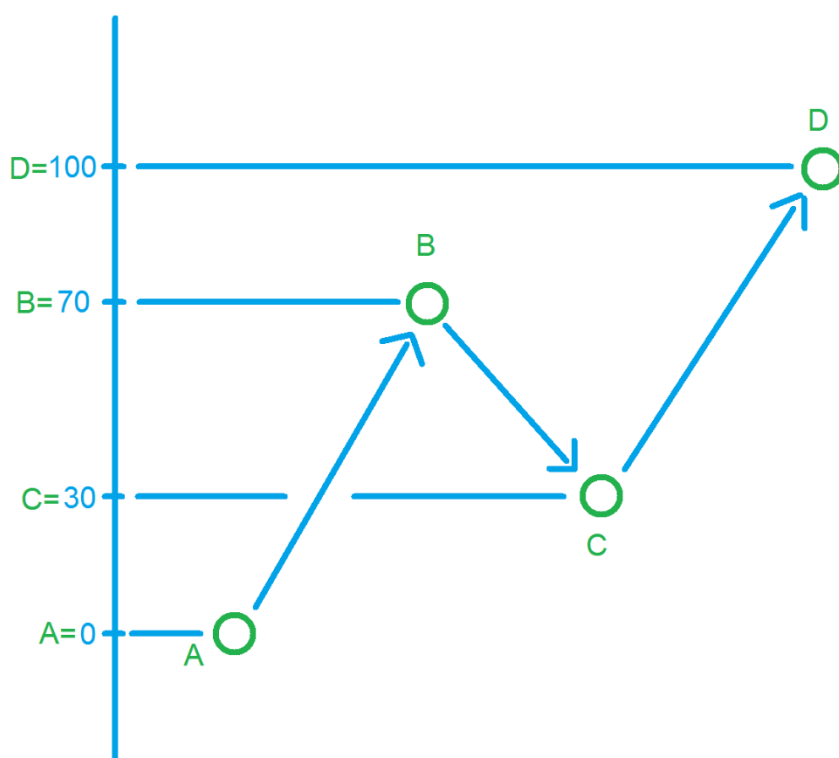


Obrázok č. 27: Príklady niektorých možných vzorov.

(Zdroj: Vlastné spracovanie)

3.1 Princíp definovania Patternu

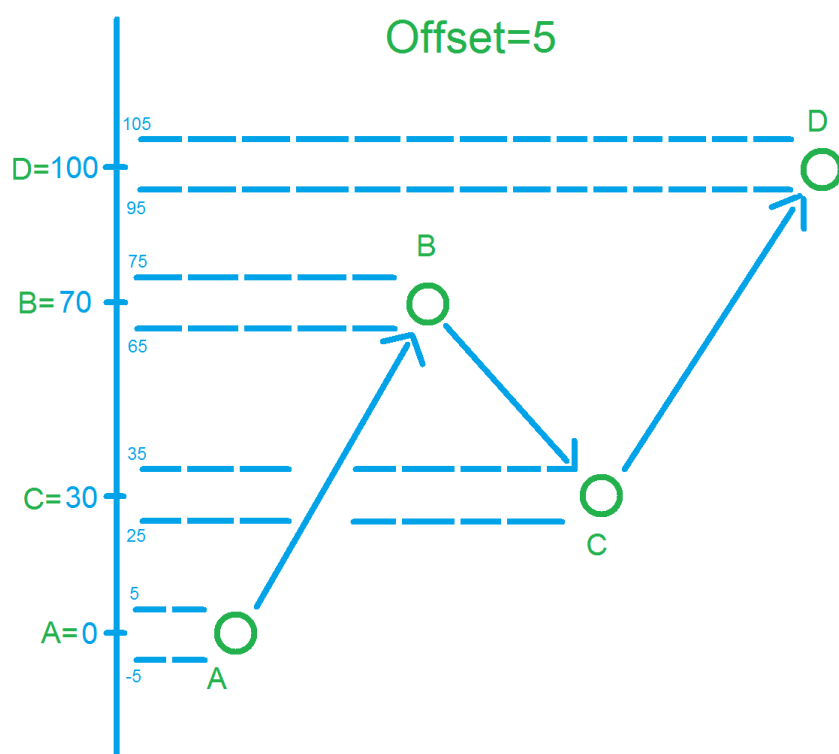
Predtým než bude možné hľadať vzory, je nutné ich definovať v kóde programu, a to tak, že každému uzlu patternu sa priradí jeho relatívna výška vzhľadom na celkovú výšku patternu. Počet uzlov v patterne je obmedzený na maximálne 9, avšak v realite mnohokrát nemá význam sledovať patterny zložené z viac ako šesť uzlov. Z obrázku nižšie možno ľahko pochopiť princíp definovania výšok jednotlivých uzlov v hľadanom patterne. V tomto prípade vzor pozostáva zo 4 uzlov, pričom najnižší bod patternu dostane hodnotu 0 a najvyšší hodnotu 100.



Obrázok č. 28: Vizuálna reprezentácia definovania uzlov v patterne.

(Zdroj: Vlastné spracovanie)

Treba však myslieť aj nato, že program bude fungovať exaktne na základe podmienok ktoré mu definujeme, a preto program disponuje definovateľnou toleranciou („offset“), ktorá zavedie miesto exaktných bodových definícií uzlov, intervalovo definované uzly. Vizuálnu reprezentáciu tohto prístupu možno vidieť v ďalšom obrázku.



Obrázok č. 29: Intervalová definícia uzlov pomocou offsetu

(Zdroj: Vlastné spracovanie)

Pokiaľ máte definované uzly a offset vášho patternu, je dôležité zdefinovať ešte ďalšie parametre ako minimálna a maximálna veľkosť patternu. Pattern definovaný v obrázkoch vyššie by sme mohli nájsť na rôznych časových rámcoch, a teda v rôznych veľkostiach alebo aj na jednom časovom rámci ale pozostával by z väčších alebo z menších sviečok. Preto je nutné definovať aspoň jeho minimálnu veľkosť. Táto definícia je však exaktná a treba s ňou citlivo narábať, inak sa môže stať, že váš pattern, program odignoruje a to len na základe drobných odchýlok v požadovanej veľkosti. Veľkosťou patternu sa myslí, rozdiel medzi low hodnotou najnižšieho bodu patternu a high najvyššieho bodu patternu. Tieto hodnoty udávame v „pip“ hodnotách ktoré sa líšia inštrument od inštrumentu, avšak je praktické ich používať, minimálne v prostredí Metatraderu. Možno ich ľahko zmerať z grafu bez potreby počítania na tisíciny jednotiek ak by sme sa bavili o menových pároch. Na ďalšom obrázku možno vidieť ako jednoducho takéto meranie prebieha a ako ľahko vieme získať predstavu o tom ako veľký pattern by sme preferovali v našej stratégii.



Obrázok č. 30: Meranie výšky patternu v „pip“ hodnotách

(Zdroj: Vlastné spracovanie)

3.2 Princíp fungovania algoritmu

Keďže už viete ako by ste efektívne zadefinovali pattern ktorý hľadáte, zostáva vám zistiť, ako túto požiadavku algoritmicke vyriešiť. Pri problematike technických analýz, možno použiť voľne dostupné indikátory z ktorých možno cez API získavať ich hodnoty a nastavovať ich podľa vlastnej potreby. Pre identifikáciu „support“ a „resistance“ zón ma konkrétne zaujal jeden, menom ZigZag, ktorý možno vidieť na obrázku nižšie.



Obrázok č. 31: Indikátor ZigZag v MetaTrader 4

(Zdroj: Vlastné spracovanie)

Pri pokúšaní sa spätne skonštruovať tento indikátor, narážam na mnohé premenné, ktoré sa vo výslednom programe odzrkadlia ďalšími nastavitelnými parametrami. Pre začiatok však musím zistiť ako principiálne identifikovať uzly ktoré ukazuje tento indikátor. Ak sa na chvíľu zadívate na tento indikátor zistíme, že uzly ktoré identifikuje sú najvyššími alebo najnižšími bodmi za istú periódu sviečok, viď obrázok nižšie.



(Zdroj: Vlastné spracovanie)

Indikátor tiež možno interpretovať nasledovne podľa ďalšieho obrázku, ak sa držím myšlienky, že jednotlivé uzly sú najvyššími alebo najnižšími bodmi za istý počet sviečok, mal by som si definovať, koľko týchto sviečok musí byť potreba aby sa sviečka mohla nazvať uzlom. Pre tento prípad môžeme povedať, že aby bola sviečka považovaná za vrchol, tak musí byť najvyššia za posledných aspoň 5 sviečok z ľavej strany a 3 sviečky z pravej strany, Týmže bod neprehlásime za uzol do chvíle, než pribudnú ďalšie 3 sviečky však nemožno nijak obísť. Žiaden indikátor nevidí do budúcnosti, a teda ak by ste napríklad chceli vždy, keď objavíte vrchol, ísť do short pozície, musíte počkať pár sviečok, než sa vám tento vrchol objaví. U menších vrcholov to ale môže znamenať, že o niekoľko sviečok bude po investičnej príležitosti, preto to, koľko sviečok z pravej a z ľavej strany bude nutné na uznanie uzlu spravím ako voliteľné parametre, a to zvlášť pre pravú a pre ľavú stranu. Praktickou ukážkou ako teda funguje indikátor ZigZag je nasledovný obrázok, ak si predstavíte, že sa nachádzame v bode A_1 , tak platí, že ešte nepoznáte sviečky ktoré pribudnú od tohto bodu smerom doprava, Indikátor teda bod A_1 vôbec nevykreslí ako uzol. To znamená, že do vtedy indikátor zobrazuje iba naposledy vykreslený „bottom“, znázornený modrou farbou, ako doteraz najnižší bod. Ak však počkáme na uzavretie ďalších troch sviečok, bod A_1 nám indikátor uzná, pretože už splňuje podmienku najvyššieho bodu aj z pravej strany. V momente keď máme vykreslený nový „top“, indikátor by mal za ďalší uzol hľadať „bottom“ aby sa nám striedali „topy“ a „bottomy“. Čo sa však stane ak indikátor nájde miesto „bottom“ ešte vyšší „top“ ? Indikátor v tomto prípade nečaká, maže bod A_1 a presúva ho na pozíciu najnovšej sviečky bez toho aby kontroloval podmienku pravej strany. To znamená, že body ktoré indikátor vykreslí nie sú definitívne pokiaľ indikátor neobjaví po „tope“ „bottom“ alebo po „bottom“ „top“. V obrázku môžeme sledovať, že bod A sa niekoľkokrát premiestnil, reálne dokonca viac ako 5krát, ak sa zadívame na sviečky tak si môžeme v niektorých prípadoch všimnúť že vtedy aktuálny bod A_x bol prevýšený niektorými sviečkami, avšak kvôli prehľadnosti som sa ich rozhodol neznázorniť, pretože sa jednalo o drobné rozdiely vo výške jednotlivých sviečok. Mnoho ľudí si však túto vlastnosť neuvedomuje a považuje indikátor za perfektný pretože ak sa dívame na historické dáta, tak nám ukazuje perfektné „tops“ a „bottoms“, pričom sa klamlivo zdá, že dokázal perfektne identifikovať obrátové body trendu. Preto v mojej verzii indikátoru budem podmienky nastavovať podľa vlastných potrieb a musíme prijať fakt, že aby sme

identifikovali uzol, musíme sa vzdať naivnej myšlienky, že nejaký indikátor dokáže nájsť zlomový bod trendu a pracovať s údajmi ktoré sú nám dostupné, a to čo najlepšie. Všimnite si teda ešte opačnú situáciu a to pri bode B, pre predstavu bod B₁ bol vykreslený indikátorom, až po tretej zelenej sviečke ktorá za ním nasleduje, to znamená, že investičná príležitosť môže byť už z polovice premeškaná, a teda ak by sme rozpoznanie tohto uzlu považovali za nákupný signál, môžeme vidieť, že na nasledujúcich sviečkach sa ukázalo že sa jednalo len o korekciu a dokonca sme prehľadli dno a bod B sa presunul na B₂. Prierazom k tejto hodnote by bolo múdre uzavrieť náš prípadný obchod aj keď so stratou. Nemôžeme totiž tušiť kam až sa cena môže prepadnúť, avšak v tomto prípade sa stal presný opak, takýmto krokovým sledovaním možno pochopiť, že obchodovanie je velice zradnou záležitosťou a, že žiaden indikátor, nech je akokoľvek na oko lákavý nevie predvídať budúcnosť len tak, bez nejakej stratégie ktorá na ňom stavia. Z tejto ukážky nevyspytateľnosti trhu si tiež možno odniesť to, že výhodou algoritmického obchodovania je možnosť eliminácie ľudského faktoru a to v tej forme, že človek za správaním sa trhu hľadá často „blafy“ veľkých hráčov, ktoré sa samozrejme reálne dejú, avšak u ľudského obchodníka spôsobujú nedodržiavanie svojej vlastnej stratégie. Taktiež mnoho obchodníkov sa domnieva, že dokážu svojou intuíciou nájsť zlomové body v trende, pričom algoritmicke si môžu pomocou backtestu veľmi ľahko otestovať svoje hypotézy bez toho aby museli platiť reálne peniaze alebo čakať na samotný budúci vývoj ceny.



Obrázok č. 33: Znáznornenie postupného prekresľovania indikátoru ZigZag

(Zdroj: Vlastné spracovanie)

Tým že máme vysvetlený princíp fungovania indikátoru na ktorom budem stavať svoje technické analýzy, možno začať premýšľať nad tým ako bude fungovať rozpoznávanie patternov ktoré sme si zadefinovali. V tomto momente už vieme, že naše patterny sú definované jednotlivými uzlami a ich relatívnymi výškami vzhľadom na celkovú výšku patternu. Ďalej nám tu figuruje offset, ktorý značí istú toleranciu odchýlok pri rozpoznávaní patternov. Dáta ktoré dostaneme od indikátoru budú uložené v štruktúre menom pole, táto štruktúra dovoľuje ukladať dáta sekvenčne a udržiavať ich v poradí, teda aj prístup k týmto dátam bude možné riadiť inkrementujúcou sa premennou, v praxi to znamená, že k uzlom nášho indikátoru vieme pristupovať jednoduchými dotazmi na ich číselné indexy v poli. Samotné rozpoznávanie patternu bude spočívať v tom, že vezmeme isté indexy z poľa vytvoreného indikátorom a prirovnáme ich k nami hľadanému patternu, to znamená, že budeme zisťovať či niektoré z uzlov v poli splňujú podmienky definované pre náš pattern. Avšak tým, že náš pattern môže pozostávať z rôzneho počtu uzlov, je nutné uspošobiť kód jeho dynamickým vlastnostiam, nesmieme ale zabúdať aj nato, že tento program bude schopný sám po nájdení patternu vyvolať predom definovanú akciu a teda môže sa jednať o market príkazy alebo o príkazy typu stop/limit, preto v programe budú figurovať ako voliteľné premenné na nastavenie targetov ako stop loss a take profit, pre príkazy stop a limit však je nutné pridať nastavenia expirácie príkazu a jeho otváraciu cenu. Samozrejme to neznie ako problém, v programe iba bude figurovať viac inicializačných premenných. Tieto hodnoty bude nutné uspošobiť veľkosti patternu ktorý hľadáme, preto targetové hodnoty všeobecne a otváracie ceny pre stop/limit príkazy budú udávané v percentách výšky patternu, ako príklad môžeme uviesť nasledovný obrázok kde si môžeme všimnúť „Double Top“ pattern pre ktorý som nadefinoval okamžitý nákup po jeho rozpoznaní, stojí za povšimnutie, že možno definovať hodnoty aj mimo rozsahu 0 až 100 pre samotné obchody, v tomto prípade pattern zafungoval presne podľa očakávaní, taktiež netreba zabúdať, že sa síce jednalo o market order avšak bol exekúovaný až niekoľko sviečok za patternom a to z toho dôvodu, že samotný pattern nebol rozpoznaný skôr, to pramení z princípu fungovania nášho indikátoru, ktorý kontroloval podmienku troch sviečok z pravej strany aby posledný bod uznal za uzol.

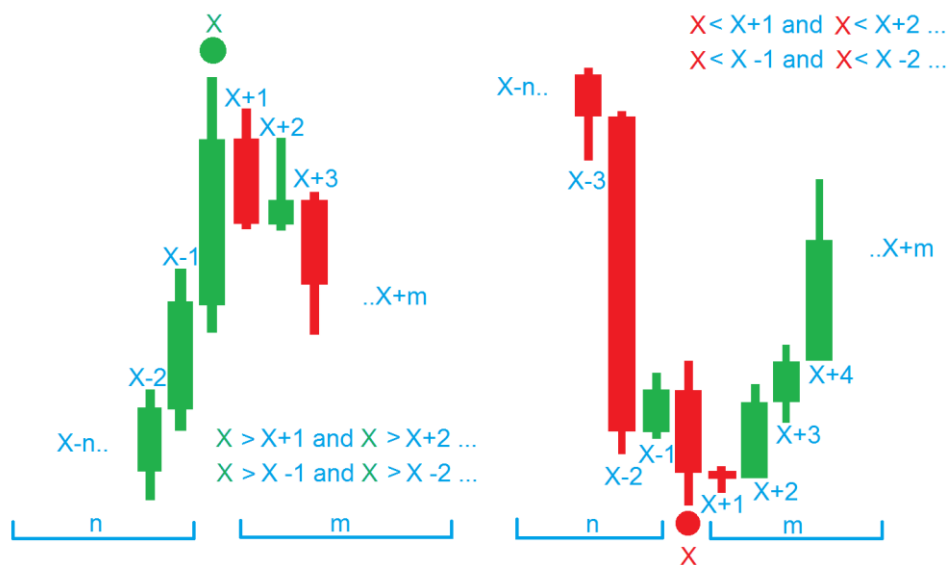


Obrázok č. 34: Príklad patternu v praxi

(Zdroj: Vlastné spracovanie)

3.3 Implementácia algoritmu

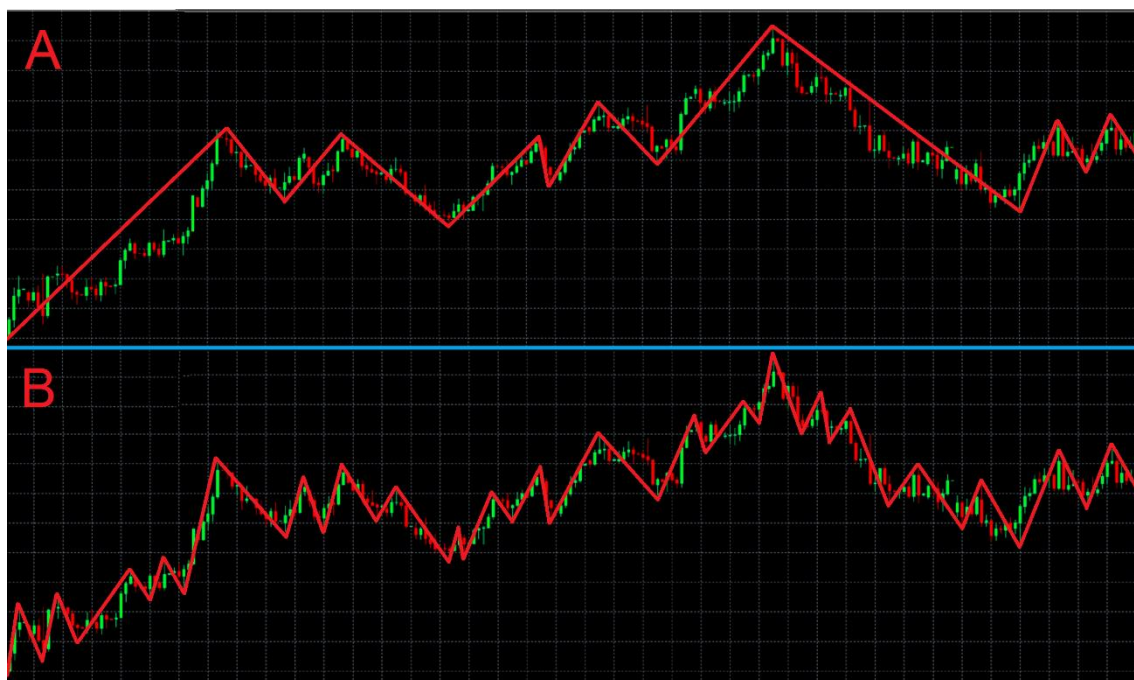
Dôležitým prvkom algoritmu je práve indikátor ktorý bude schopný rozpoznávať jednotlivé uzly v grafe v čo najkratšom čase. Ako princíp pri rozpoznávaní uzlov si vytvorím podmienky ktoré budú kontrolovať „high“ a „low“ parametre sviečky a prirovnávať ich ku sviečkam v ich bezprostrednom okolí. Teda ako príklad si vezmem jednu sviečku a pozrieme sa či jej susedné sviečky sú od nej vyššie alebo nižšie, ak bude platiť že všetky jej susedné sviečky sú nižšie, možno prehlásiť danú sviečku za uzol, v tomto prípade vrchol alebo rezistenciu, v opačnom prípade, ak low sviečky je tým najnižším z pomedzi jej susedov, možno vyhlásiť, že sa jedná o isté lokálne minimum a označiť túto sviečku za uzol alebo support zónu. Počet susedných sviečok ktoré budú skúmané necháme ako globálnu premennú, a to samostatne pre počet sviečok z ľavej strany a samostatne pre počet z pravej strany. Tieto hodnoty si bude môcť užívateľ nastaviť podľa seba. V ďalšom obrázku možno vidieť ako vyzerajú sviečky keď spĺňujú tieto podmienky



Obrázok č. 35: Rozpoznávanie uzlov algoritmom ilustrované

(Zdroj: Vlastné spracovanie)

Taktiež je veľmi dôležité aby algoritmus dokázal meniť parametre n a m zobrazené v predošlom obrázku, pretože ich hodnoty zásadne menia output. Na ďalšom obrázku možno pozorovať rovnaký graf avšak s použitím rôznych nastavení, v prípade verzie A som mal požiadavku na parametre m a n väčšiu než v prípade B, to znamená, že v prípade A musela byť kontrolovaná sviečka najvyššou z pomedzi väčšieho množstva susedov než v prípade B, kde stačilo prevýšiť len dvoch susedov.



Obrázok č. 36: Porovnanie dvoch rôznych nastavení môjho indikátoru

(Zdroj: Vlastné spracovanie)

3.3.1 Implementácia podporného indikátoru

Implementáciu indikátoru možno popísať nasledovne, zadeklarujem si objektovú štruktúru menom `uzol`, kde budem do atribútu ukladať poradové čísla sviečok ktoré vyhodnotím ako uzol, do druhého atribútu budem vkladať informáciu, či daný uzol symbolizuje „top“ alebo „bottom“ v binárnej podobe hodnôt 1 alebo 0. Následne zadeklarujem globálne pole ktorého bunky budú typovo zodpovedať uzlom ktoré som práve popísal. To znamená, že máme definované pole do ktorého možno vkladať, odoberať, prepisovať, listovať alebo len čítať hodnoty. Popri chode programu toto pole bude postupne chronologicky vyplňané uzlami. V nasledujúcom obrázku môžete vidieť reálnu deklarácie tejto štruktúry.

```
25 struct uzol
26 {
27     bool        top;
28     int         candleIndex;
29 };
30
31 uzol arr[size];
```

Obrázok č. 37: Implementácia poľa uzlov

(Zdroj: Vlastné spracovanie)

Samotnú procedúru využívajúcu túto štruktúru by som mohol popísať nasledovne. Procedúra sa privolá s nasledujúcimi povinnými argumentami, to sú počty sviečok z ľavej a z pravej strany ktoré sa budú kontrolovať ako podmienka uzlu, tretím argumentom je celkový počet sviečok, ktoré má procedúra skontrolovať. Procedúra začína od najstaršej sviečky a postupne sa prepracováva až k najnovším dátam, v tomto poradí sa následne ukladajú aj uzly, to znamená že v poli uzlov na najnižšom indexe bude figurovať najstarší uzol, preto je vhodné tento parameter voliť rozumne aby sme sa nezaoberali zbytočne starými sviečkami. Najmä pri backteste stratégie, by nevhodné nastavenie mohlo mať za následok vysokú výpočtovú zložitosť, a teda zvýšenú časovú náročnosť. To znamená, že ak hľadáme malé patterny, tak nie je dôvod tomuto nastaveniu priradovať vysoké hodnoty (30+), avšak ak hľadáme väčšie patterny, môže prísť vhod aj nastavenie 50+. Malo by byť sprevádzané aj zväčšenými hodnotami podmienok ľavej a pravej strany a to z dôvod takého, že ak by sa tieto hodnoty nechali malé, tak nájdeme viacej uzlov a teda patterny samotné budú tiež menšie, vo výsledku teda treba všetky tri

parametre voliť na základe vašich cieľov, podmienku zľava a sprava možno voliť samostatne a teda na ľavej strane môže figurovať vysoké číslo (napr. 20) a na pravej strane môže byť kľudne menšie (napr. 2), to preto, že po prípadnom rozpoznaní patternu nám ubehne len toľko sviečok koľko sme si nastavili na podmienke sprava. Na nasledujúcom obrázku možno vidieť časť kódu tohto indikátora. Obsahuje ešte štvrtý argument menom „arrows“ ktorý slúži len pre vizualizáciu v grafe (ako on/off spínač), preto sa celý kód delí na dve podobné varianty a to variantu s vykresľovaním a bez. V tele procedúry sa cyklom privoláva vyhodnocovanie jednotlivých sviečok a ich prípadné zapisovanie do poľa uzlov na základe outputu podporných funkcií.

```

863 void ZigZagFull(int oldest,int precL, int precR,int arrows){
864     arrClear();
865     int i=oldest-precL;
866     int output;
867     if(arrows==0){
868         while(i>=precR){
869             output=CheckCandleQuick(i,precL,precR);
870             if(output!=-1){
871                 arrAdd(output,i);
872             }
873             i--;
874         }
875     }
876     if(arrows==1){
877         while(i>=precR){
878             output=CheckCandleQuick(i,precL,precR);
879             if(output!=-1){
880                 arrAdd(output,i);
881             }
882             i--;
883         }
884         graphArrows();
885         //zdraw();
886     }
887 }

```

Obrázok č. 38: Hlavná procedúra podporného indikátora

(Zdroj: Vlastné spracovanie)

Samotné vyhodnocovanie sviečok je riešené podpornou funkciou ktorá dostáva ako argumenty podmienky ľavej a pravej strany a ešte poradové číslo sviečky ktorú má vyhodnotiť. Samotné vyhodnocovanie je principiálne banálne, delí sa na dva cykly, jeden pre vyhodnotenie ľavej strany od skúmanej sviečky a v druhom cykle strany pravej, vyhodnocovanie porovnáva teda podľa potreby high alebo low hodnoty sviečok v rozsahoch stanovených argumentami, následne tieto cykly inkrementujú počítadlá ktoré sa na konci funkcie vyhodnotia a na základe nich, funkcia odpovie procedúre ktorá ju volala hodnotami 1/ 0/ -1 ktoré reprezentujú nájdenie topu/bottomu/ničoho. V nasledujúcom obrázku možno vidieť ako sú tieto podmienky implementované vo funkcii.

```

889 int CheckCandleQuick(int candle, int precL, int precR) {
890     if (candle < precR) { return -1; }
891     int outputT=0;
892     int outputB=0;
893     int i=1;
894     while (i <= precR) {
895         if (iHigh(Symbol(), tf, candle) > iHigh(Symbol(), tf, candle-i)) {
896             outputT++;
897         }
898         if (iLow(Symbol(), tf, candle) < iLow(Symbol(), tf, candle-i)) {
899             outputB++;
900         }
901         i++;
902     }
903     i=1;
904     while (i <= precL) {
905         if (iHigh(Symbol(), tf, candle) >= iHigh(Symbol(), tf, candle+i)) {
906             outputT++;
907         }
908         if (iLow(Symbol(), tf, candle) <= iLow(Symbol(), tf, candle+i)) {
909             outputB++;
910         }
911         i++;
912     }
913     if ( (outputT==precL+precR && outputB==precL+precR) || (outputT!=precL+precR && outputB!=precL+precR) ) {
914         return -1;
915     }
916     if (outputB==precL+precR) {
917         return 0;
918     }
919     if (outputT==precL+precR) {
920         return 1;
921     }
922 }

```

Obrázok č. 39: Implementácia funkcie vyhodnocujúcej sviečky za uzly

(Zdroj: Vlastné spracovanie)

V prípade keď funkcia signalizuje nájdenie nového uzlu, procedúra volá funkciu ktorá pridá nový uzol do poľa uzlov, avšak z princípu fungovania indikátora vieme, že dva rovnaké vrcholy po sebe mať nechceme, to znamená, že po „top“ chceme mať uložený „bottom“ a v takomto striedavom ukladaní pokračovať až neskontrolujeme všetky sviečky ktoré sme si zvolili. Z tohto princípu sa pridávanie nových uzlov jemne komplikuje, avšak pravidlo pridávania je jasné a nasledovné, pokiaľ máme naposledy uložený „top“ a narazíme na ďalší „top“, ktorý by splňoval podmienku uzlu, tieto dva „tops“ spolu porovnáme a necháme si ten vyšší z nich, to preto, že nás zaujímajú tie najvyššie vrcholy, niekoľko obrázkov späť sme mohli pozorovať, že pri vykresľovaní indikátoru ZigZag, sa nám bod A niekoľkokrát posunul na vyššiu hodnotu než sa ustálil na najvyššom bode. Podobný princíp teda platí aj u hľadani najnižších bodov, pokiaľ sa nám vyskytnú dva „bottoms“ za sebou, ponechávame si ten ktorý je nižší a to znova preto, že nás zaujímajú najvyššie a najnižšie body grafu a nie korekcie ktoré nastávajú cestou hore alebo dolu. V nasledujúcom obrázku možno vidieť ako je štruktúrovaný kód a formulované podmienky tejto podpornej procedúry, tiež možno pozorovať volanie funkcií arrLen a arrLastWrite, tie tu vystupujú preto, že je nutné si dohľadať poradové čísla (indexy) posledných hodnôt v poli uzlov a tiež občas je nutné pristupovať k celkovému počtu prvkov a inokedy zas pristúpiť k indexu posledného zapísaného uzlu.


```

923 void arrAdd(bool newT,int newC){
924     int arrLN = arrLen();
925     int arrLW = arrLastWrite();
926     bool lastT = arr[arrLW].top;
927     int lastC = arr[arrLW].candleIndex;
928     double newP;
929     double lastP;
930     if(arrLN>0){ //THERE HAS TO BE SOMETHIN ALREADY IN ARRAY TO COMPARE TO..
931         if(newT==lastT){ //IF TWO IN ROW (tops/Bottoms)
932             //Alert(TimeToString(iTime(0,0,newC))+" "+newT+" TWO in ROW" );
933             if(lastT==1){ // IF TWO TOPS
934                 //Alert(TimeToString(iTime(0,0,newC))+" 1 Two Tops");
935                 newP = iHigh(Symbol(),tf,newC); //SET PRICE
936                 lastP = iHigh(Symbol(),tf,lastC); //SET PRICE
937                 if(newP > lastP){ //HIGHER PRICE WINS
938                     arr[arrLW].candleIndex = newC;//OVERWRITTING not adding new
939                 }
940             }
941             if(lastT==0){ // IF TWO BOTTOMS
942                 //Alert(TimeToString(iTime(0,0,newC))+" "+newT+" Two Bottoms");
943                 newP = iLow(Symbol(),tf,newC); //SET PRICE
944                 lastP = iLow(Symbol(),tf,lastC); //SET PRICE
945                 if(newP < lastP){ //LOWER PRICE WINS
946                     arr[arrLW].candleIndex = newC;//OVERWRITTING not adding new
947                 }
948             }
949         }
950         if(newT!=lastT){ //IF NEW IS DIFFERENT FROM PREVIOUS
951             //Alert(TimeToString(iTime(0,0,newC))+" "+newT+" Different tops");
952             arr[arrLN].candleIndex = newC; //ADDING NEW ELEMENT TO THE ARRAY
953             arr[arrLN].top = newT; // /// ///
954         }
955     }
956     else{ //IF EMPTY ARRAY
957         //Alert(TimeToString(iTime(0,0,newC))+" "+newT+" First Top");
958         arr[arrLN].candleIndex = newC; //ADDING NEW ELEMENT TO THE ARRAY
959         arr[arrLN].top = newT; // /// ///
960     }
961 }

```

Obrázok č. 40: Implementácia pridávania nových uzlov do poľa

(Zdroj: Vlastné spracovanie)

V momente ako máme pole naplnené a nezostávajú nám ďalšie sviečky ktoré by sme preskúmali, je práca indikátoru dokončená a voláme procedúru na vyhodnotenie patternu ktorý sme si definovali. Táto zostava prebieha pri každom uzavretí sviečky zvoleného time frame, teda sa nám nanovo vypočíta indikátor ktorý naplní pole uzlov a následne sa spustí rozpoznávanie patternov na tomto poli.

3.3.2 Definícia patternu v kóde

Samotné princípy definovanie patternu sme si už vysvetlili a v nasledujúcom obrázku môžeme pozorovať ako sa tieto paternity deklarujú v kóde a akými dátovými typmi a štruktúrami sú definované. Môžeme pozorovať v ľavej časti obrázka aké atribúty drží štruktúra pattern a pod ňou môžeme vidieť deklaráciu poľa týchto patternov, to z toho dôvodu, že rozpoznané paternity budeme udržiavať po celú dĺžku bežania programu, na rozdiel od poľa uzlov ktoré sa bude aktualizovať každou novou sviečkou, je tak preto, že ak by sme našli jeden pattern viackrát za sebou, mohlo by hroziť že na ňom spustíme tiež dvakrát za sebou obchodný príkaz, taktiež zo štatistických dôvodov nás môže zaujímať koľko patternov sme našli a teda duplicity by tieto dáta mohli znehodnotiť. Od riadku 55

nižšie môžeme vidieť definíciu ďalšej štruktúry, tá sa už týka hodnôt nami zadávaných a to samotných uzlov patternu ktorý hľadáme, môžeme vidieť atribút „top“ ktorý značí o aký druh vrcholu sa jedná a atribút percentil, ktorý značí jednotlivú výšku týchto bodov od 0 do 100, následne sme definovali pole s 11 prvkami, typu štruktúra ktorú sme si teraz popisovali. Na pravej strane môžeme vidieť premenné a hodnoty ktorými sa definuje náš pattern, ich úloha je teraz už zjavne jasná, stojí za povšimnutie definícia jednotlivých uzlov, kde v podstate iba zadáme do poľa ako za sebou nasledujú jednotlivé uzly a aké sú ich výšky, analógia s predošlými ilustráciami je v tomto momente asi zjavná.

```

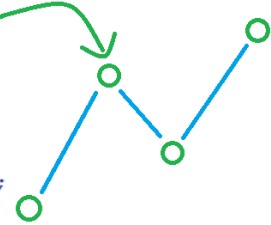
39 struct pattern
40 {
41     long      startCandle;
42     long      lastCandle;
43     double    zeroPercentPrice;
44     double    patternSize;
45     int       len;
46
47 };
48
49
50
51 pattern arrP[size];
52
53 //-----
54
55 struct patternAttribute
56 {
57     int top;
58     int percentil;
59 };
60
61
62 patternAttribute arrAt[10];

```

```

223 off=5;           //offset for patterns
224 minSize = 1*Point; //min pattern size: 0.002
225 maxSize =9999 ;    //max pattern size: 0,004
226
227
228
229
230
231 arrAt[0].percentil=0;
232 arrAt[0].top=0;
233
234
235 arrAt[1].percentil=70;
236 arrAt[1].top=1;
237
238 arrAt[2].percentil=30;
239 arrAt[2].top=0;
240
241 arrAt[3].percentil=100;
242 arrAt[3].top=1;
243
244 //-----

```



Obrázok č. 41: Implementácia definície patternu

(Zdroj: Vlastné spracovanie)

3.3.3 Pribeh procesu identifikácie patternu

V počiatku procedúry hľadajúcej tento zvolený pattern, je nutné skontrolovať a vyhodnotiť zadané údaje, program si uloží do premenných informácie o tom, na ktorých indexoch má očakávať najnižší a najvyšší bod patternu, v tomto prípade sú to indexy 0 a 3, to znamená, že prvý uzol patternu by mal byť najnižším a naopak štvrtý uzol najvyšším. Z týchto okrajových hodnôt sa následne bude vypočítavať samotná výška patternu a na základe ktorej sa budú dorátavať aj ostatné uzly. Procedúra taktiež využíva mnoho podporných funkcií ako sme mohli vidieť pre implementáciu indikátora použitie podporných funkcií k sledovaniu zaplnenia poľa alebo k prístupu k indexu najčerstvejšej hodnoty, analogicky budeme používať podobné funkcie na prehľadávanie polí aj pri hľadaní patternu. Samotné hľadanie patternu prebieha nasledovne, bude prechádzať od

konca poľom ktoré mu pripravil indikátor, v tomto poli sa nachádzajú uzly, pričom istá postupnosť týchto uzlov môže tvoriť pattern ktorý hľadáme. Pokiaľ hľadáme pattern tvorený štyrmi uzlami, procedúra sa bude dívať na štvorice bodov v poli a porovnávať ich relatívne vzájomné polohy vzhľadom na seba a to na základe podmienok ktoré sme si stanovili, samotné uzly v poli sú reprezentované indexami sviečok, preto je nutné dotázať sa na ceny skrývajúce sa pod týmito indexami a z nich dopočítať vzájomné polohy jednotlivých uzlov, tieto vzájomné polohy následne porovnať s tými ktoré by sme očakávali v našom patterne. V praxi priebeh programu spočíva v cyklickom prechode po celom poli uzlov, kde sa berú n-tice uzlov (na základe príkladu z obrázkov, n-tica v danom prípade predstavovala štvoricu), na týchto n-ticiach sa robí analýza pomocou vnoreného cyklu ktorý vezme jednotlivé uzly z vybranej n-tice a prirovnáva ich k tým ktoré by sme očakávali. S prítomnosťou offsetu nám hodnoty nemusia súhlasiť úplne presne, musia sa len vtesnať do definovanej tolerancie, ak uzol sedí v definovanej tolerancii, cyklus sa pohne k ďalšiemu uzlu pre ktorý znova vypočíta očakávaný interval a porovná ho s polohou ktorú zastupuje tento uzol vo vybranej n-tici, ak by uzol nevyhovoval podmienkam, cyklus sa zastavuje a vraciame sa k výberu novej n-tice, nepreskakujeme však celú n-ticu bodov ktoré sme skúmali ale sa posúvame len o jeden uzol ďalej, teda nová vybraná n-tica bude disponovať iba 1 novým uzlom pričom najstarší uzol sa zahodí, ako však vieme pole uzlov je plnené striedajúcimi sa „tops“ a „bottoms“, teda keď hľadáme pattern ktorý sa začína „bottom“ uzlom, posun na ďalšiu n-ticu môže znamenať, že hneď prvý uzol bude miesto „bottom“ práve „top“, ani v takomto prípade sa však nič nedeje, zanorený cyklus vyskočí hneď v prvej podmienke pri kontrolovaní prvého uzla a to preto, že sa kontroluje nielen cenová hladina ale aj to či daný bod predstavuje rezistenciu alebo support zónu, v tom prípade sa teda posunie hlavný cyklus znova na ďalšiu n-ticu uzlov, znova však len o jeden uzol. Takýto prechod poľom môže byť výpočtovo náročný ak si zvolíme, že naše pole uzlov budeme plniť príliš hlboko do histórie. V nasledujúcom obrázku môžeme pozorovať kus vnútorného cyklu kde sa kontrolujú samotné uzly n-tíc.

```

417 while(j<arrAtLN){
418     //-----
419     jc=arr[i+j].candleIndex;
420     jt=arr[i+j].top;
421     //Alert("in cycle "+jc+" jt="+jt+" len="+arrAt[j].len);
422     if(jt==arrAt[j].top){
423
424         if(jt==1){
425             currentCandlePrice=iHigh(0,0,jc);
426         }
427         else{
428             currentCandlePrice=iLow(0,0,jc);
429         }
430         offsetSize=offset*patternSize;
431         //Alert("OffsetP: "+offsetP+" offset: "+offset+" minP: "+minP+" maxP: "+maxP+" minP: "+minP);
432         if(arrAt[j].percentil>=1){ //percentil 1 - 5
433             expectedPercentil=NormalizeDouble(arrAt[j].percentil,5)/100;
434         }
435
436         if(arrAt[j].percentil==0){
437             expectedPrice = minP;
438         }
439         else{
440             expectedPrice = minP+(patternSize*expectedPercentil);
441         }
442         currentCandleTopBorder = expectedPrice+offsetSize;
443         currentCandleBottomBorder = expectedPrice-offsetSize;
444
445         //Alert(" "+jc+" EXPECT $n.r.b.bh: "+expectedPercentil+" "+expectedPrice+" "+cur
446         if(currentCandlePrice<=currentCandleTopBorder &&
447            currentCandlePrice>=currentCandleBottomBorder){
448
449             resultForCurrentIteration++;
450         }
451         //Alert(" Given $currentCandlePrice:
452         //Alert(" passed "+ " +jt+" == "+arrAt[j].top+" PS,CE,CBB "+DoubleToStr(maxP
453         //Alert(" "+jc+" JC: "+jc+" JT: "+jt+" want "+arrAt[j].top
454     }
455     else{
456         //Alert(" else "+jt+" == "+arrAt[j].top);
457         j=arrAtLN;
458     }
459     //-----
460     j++;

```

Kontrola či vybraný uzol je rovnakého typu ako hľadaný.

Podľa toho či máme top alebo bottom vybereme High alebo Low hodnotu pre počítanie.

Prepočet očakávaných hodnôt pre vyhodnocovanie intervalu v ktorom by sa mala nachádzať cena.

Porovnanie ceny vybraného uzla s očakávanými cenami podľa užívateľom zadanych pravidiel hľadaného patternu.

Obrázok č. 42: Implementácia vnútorného cyklu porovnávania uzlov indikátora s hľadaným patternom

(Zdroj: Vlastné spracovanie)

V prípade, že vybraná n-tica splňuje podmienky nášho patternu, je táto n-tica zapamätaná do globálnej pamäti aby pri opätovnom prechode nezapočítala opätovné nájdenie tohto patternu, zapamätanie patternu s ľubovoľnou dĺžkou je riešené uložením si jedinečných parametrov tohto patternu ako je cena u jeho najnižšieho uzlu, celková výška patternu vyjadrená v cenových jednotkách alebo aj dĺžka patternu vyjadrená v počte sviečok. Teda pri nájdení patternu sa zavolá funkcia saveP(pattern) ktorá tieto jedinečné údaje porovná s tými ktoré už existujú, existujúce údaje predstavujú predošlé nájdené patterny, pokiaľ sa teda jedná o nový pattern, bude uložený a bude následne spustené aj jeho vykreslenie do grafu kde sa pomocou dostupných API príkazov dajú spojiť jednotlivé uzly čiarami, na to nám stačí poznať indexy ich sviečok. V nasledujúcom obrázku možno vidieť cyklus v ktorom sa vyberajú dvojice bodov z n-tice bodov patternu a následne sú spojené čiarou, po dokončení cyklu tieto čiary vytvoria obrazec patternu v grafe.

```

534 temp_j=0;
535 temp_i=i;
536 while(temp_j<arrAtLN-1){
537
538     startI = arr[temp_i].top;
539     endI = arr[temp_i+1].top;
540
541     if(startI==1 && endI==1){ // BOTH TOPS
542         startP = iHigh(0,0,arr[temp_i].candleIndex);
543         endP = iHigh(0,0,arr[temp_i+1].candleIndex);
544     }
545
546     if(startI==0 && endI==0){ // BOTH BOTTOMS
547         startP = iLow(0,0,arr[temp_i].candleIndex);
548         endP = iLow(0,0,arr[temp_i+1].candleIndex);
549     }
550     if(startI==1 && endI==0){ // First TOP, Second BOTTOM
551         startP = iHigh(0,0,arr[temp_i].candleIndex);
552         endP = iLow(0,0,arr[temp_i+1].candleIndex);
553     }
554     if(startI==0 && endI==1){ // First Bottom, Second Top
555         startP = iLow(0,0,arr[temp_i].candleIndex);
556         endP = iHigh(0,0,arr[temp_i+1].candleIndex);
557     }
558
559     TrendLine(iTime(Symbol(),tf, arr[temp_i].candleIndex), startP, iTime(Symbol(),tf, arr[temp_i+1].candleIndex), endP);
560     temp_i++;
561     temp_j++;
562 }
563
564 //Alert("next i "+i+" ");
565 //Alert(i+"-----i= IC:"+ic+" II:"+it+"-----"+TimeToString(iTime(0,0,arr[i].candleIndex)));
566 i++;

```

Obrázok č. 43: Implementácia vykresľovania nájdeného patternu do grafu

(Zdroj: Vlastné spracovanie)

3.3.4 Zadávanie obchodných príkazov

Poslednou akciou pri nájdení patternu je možnosť otvorenia obchodu. Pokiaľ však nechceme zakaždým hľadať v kóde kam vložiť verziu platobného príkazu s predstavovanými parametrami, táto zdanlivo jednoduchá akcia je sprevádzaná veľkým množstvom parametrov ktoré musia byť presmerované do nákupného príkazu uprostred kódu. Keďže máme mnoho druhov príkazov a tie sa samozrejme líšia parametrami, musíme každému parametru definovať premennú, taktiež veľkosť nájdeného patternu sa bude líšiť v mnohých prípadoch a teda fixné stop lossy a take profity sú neprínosným riešením. Preto tieto targety bude možné vyjadriť podobne ako definovanie uzlu patternu a teda do premenných nahráme relatívnu výšku vzhľadom na veľkosť patternu. V tomto prípade sa nedržím len rozsahu 0 až 100 ale možno tieto hranice presiahnuť aj z hora aj zdola, ak nastavíte napríklad take profit hodnotu na 200, toto číslo v reálnej cene reprezentuje dvojnásobnú výšku patternu od jeho najnižšieho bodu. Samotné platobné príkazy možno kombinovať nasledovne, po nájdení patternu je možné otvoriť 3 príkazy naraz, jeden market order, jeden pending order pre Buy a jeden pre Sell (stop/limit). Vypláva to z princípu ich ovládania, sú k dispozícii niečo ako on/off spínače, kde si možno zvoliť, či použijeme v rámci market príkazov long, short alebo nič, zvolením jednej z variant sa automaticky take profit a stop loss parametre presmerujú do jej obchodného príkazu, analogicky to platí pre príkazy typu stop a limit kde si tiež volíme

medzi long alebo short variantou tohto príkazu. Ostatné parametre jej tým pádom automaticky prislúchajú. Na nasledujúcom obrázku možno vidieť ako sú implementované tieto premenné a možno vidieť on/off prepínače pre zapínanie príkazov.

```

250 //-----MARKET BUY/SELL-----
251
252 okamzityNakup=1; // buy==1 sell==2 0==nic
253
254 volumeO =1;
255 okamzitySL=-100;
256 okamzityTP=150;
257 okamzitaExpirace =1000;
258
259
260 //-----BUY LIMIT/BUY STOP-----
261
262 pendingOrderB =0; // buy limit==1 buy stop==2 0==nic
263
264 volumePB=1; //volume Pending Buy"
265 entryPriceB=150; //entry Price Buy"
266 pendingSLB =0; //pending Stop Loss Buy"
267 pendingTPB =200; //pending Take Profit Buy"
268 pendingExpiraceB =288000; //pending Expirace Buy"
269 expiraceForTriggeredBuy=288000; //expirace pro nasledny BUY
270
271 //-----SELL LIMIT/SELL STOP-----
272
273 pendingOrderS =0; //sell limit==1 sell stop==2 0==nic
274
275 volumePS=1; //volume Pending Sell"
276 entryPriceS=30; //entry Price Sell"
277 pendingSLS =100; //pending Stop Loss Sell"
278 pendingTPS =-50; //pending Take Profit Sell"
279 pendingExpiraceS =488000; //pending Expirace Sell"
280 expiraceForTriggeredSell=488000; //expirace pro nasledny sell
281
282 //=====

```

Obrázok č. 44: Implementácia spínačového ovládania obchodných príkazov

(Zdroj: Vlastné spracovanie)

3.4 Demonštrácia algoritmu

V tejto chvíli je kompletne implementovaný algoritmus a možno ho nasadiť na ľubovoľný sledovaný pattern. Podstatou tejto práce však je vytvorenie samotného algoritmu, schopného identifikovať patterny a obchodovať na základe nich, preto predvedené patterny, sú len ukážkami možností algoritmu a nie hotová obchodná stratégia založená na mnou doporučených patternoch k sledovaniu. Vytvorenie vlastného portfólia stratégií by nemalo zahrňovať jeden pattern, ktorý sa vyskytne niekoľkokrát denne s nízkou spoľahlivosťou ziskovosti, ale práve z niekoľko druhov patternov ktoré sa nemusia vyskytovať na dennej báze, ale majú vysokú spoľahlivosť. Nasledujúca ukážka zobrazuje patternu „Head&Shoulders”. Použil som nastavenie 0, 80, 65, 100, 65, 80,0 a následný market long, testoval som na Bitcoine, na hodinových sviečkach. Obrázok nižšie ukazuje situáciu kedy algoritmus identifikoval tento pattern.



Obrázok č. 45: Demonštrácia algoritmu na patterne Head&Shoulders

(Zdroj: Vlastné spracovanie)

Je dôležité si uvedomiť, že pokúšať sa nájsť jeden pattern, ktorý sa bude dať obchodovať na dennej báze je nezmyselnou predstavou, a to z niekoľkých princípov, jedným je dĺžka patternu, samotný pattern môže obsahovať desiatky sviečok, teda formovanie patternu trvá istú dobu, ďalším dôvodom je princíp popularity, ak sa jeden pattern vyskytuje príliš často, tak ho vie spozorovať každý a teda jeho spoľahlivosť by dlho nevydržala. Patternové obchodovanie by malo z pravidla obsahovať väčšie množstvo spoľahlivých, no menej frekventovaných patternov, preto predvediem drobné príklady patternov ktoré sa mne ako „ne-traderovi“ podarilo nájsť. Na ďalšom obrázku možno pozorovať pattern testovaný na Bitcoine za posledné 3 roky na denných sviečkach s 18 obchodmi a 4.21 ziskovým faktorom .



Obrázok č. 46: Pattern 70,0,100

(Zdroj: Vlastné spracovanie)

Na ďalšom obrázku možno vidieť zriedkavejší pattern menom „bull triangle“ ktorý som zachytil s malou frekvenciou na štvorhodinovom časovom rámci. Samozrejme každý pattern možno otestovať na mnoho časových rámcoch a dostať diametrálne odlišné výsledky. V tomto prípade je frekvencia výskytu príliš malá nato, aby sa dali vyvodiť závery o spoľahlivosti patternu.



Obrázok č. 47: Pattern „Bull Triangle“ 0,100,40,70,40

(Zdroj: Vlastné spracovanie)

Pri patternovom obchodovaní sa netreba obmedzovať len na jeden inštrument, možno ho aplikovať ako na kryptomeny, tak na forex a akcie, v nasledujúcom obrázku ukážem výsledok z EURUSD za 3 roky, ziskový faktor po 14 obchodoch je 4.55 (zisk/strata).



Obrázok č. 48: Pattern „Bull Double Top“ 0,80,30,100,30

(Zdroj: Vlastné spracovanie)

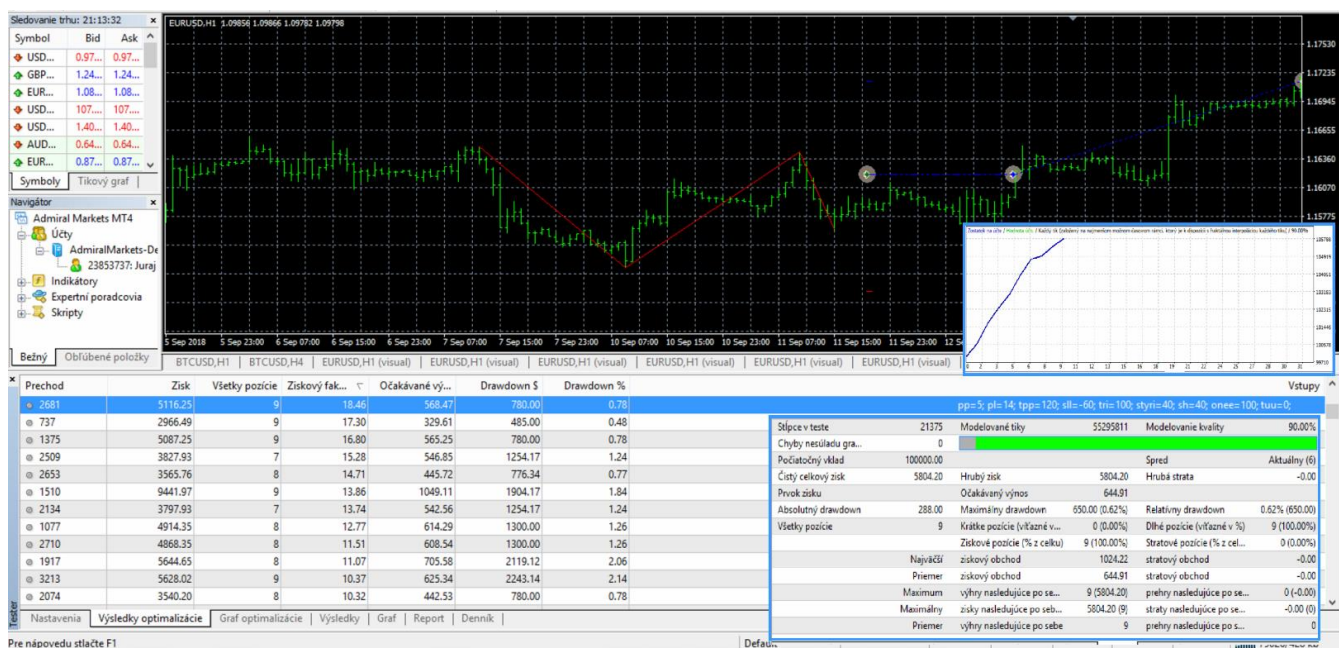
Na ďalšom obrázku možno vidieť jeden test vybraný z výsledkov optimalizácie „bullish“ patternu z EURUSD za posledné 3 roky a s umiestnený buy stop príkazom s očakávaním ďalšieho rastu ceny po rozpoznaní patternu, možno tiež vidieť, že príkaz je umiestnený dosť vysoko nato, aby ho nespustili výkyvy. Ziskový faktor po 25 obchodoch je 6.03.



Obrázok č. 49: Pattern „Bull Channel Brak“ 100,0,140,40

(Zdroj: Vlastné spracovanie)

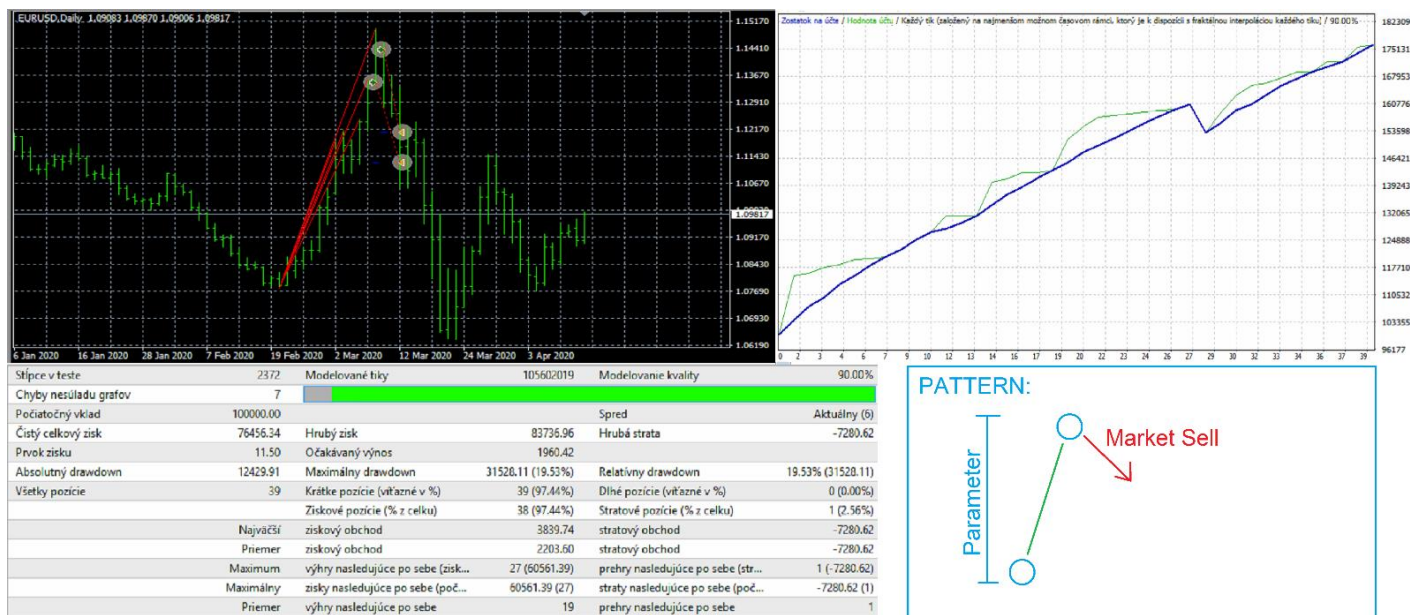
Na ďalšom obrázku sa nachádza podobný pattern ako v predošlej ukážke, Jedná sa o iný výsledok pochádzajúci z rovnakej optimalizácie, rád by som ním ukázal, aké rozdiely môže spôsobiť drobná zmena niektorých parametrov. Ako v nasledujúcom tak aj v predošlom obrázku možno vidieť mnoho výsledkov optimalizácie, kde sa rôzne menili výšky uzlov, nastavenia podporného indikátora, cieľové hodnoty ako stop loss a take profit, a tiež samotná výška, v akej bude zadáný čakajúci príkaz buy stop, ak si predstavíte, že každý z menovaných parametrov má aspoň 10 možných variant, tak nám kombinatoricky vychádzajú miliardy možných nastavení. Teda je dôležité, aby človek ktorý tieto patterny hľadá, mal aspoň základnú predstavu o tom, čo chce nájsť. V tomto prípade nedošlo k stratovej pozícii, počet obchodov sa ale zredukoval na 9, čo môže byť pre niekoho štatisticky nekonkluzívny výsledok.



Obrázok č. 50: Pattern „Channel Brak“ 100,0,100,40

(Zdroj: Vlastné spracovanie)

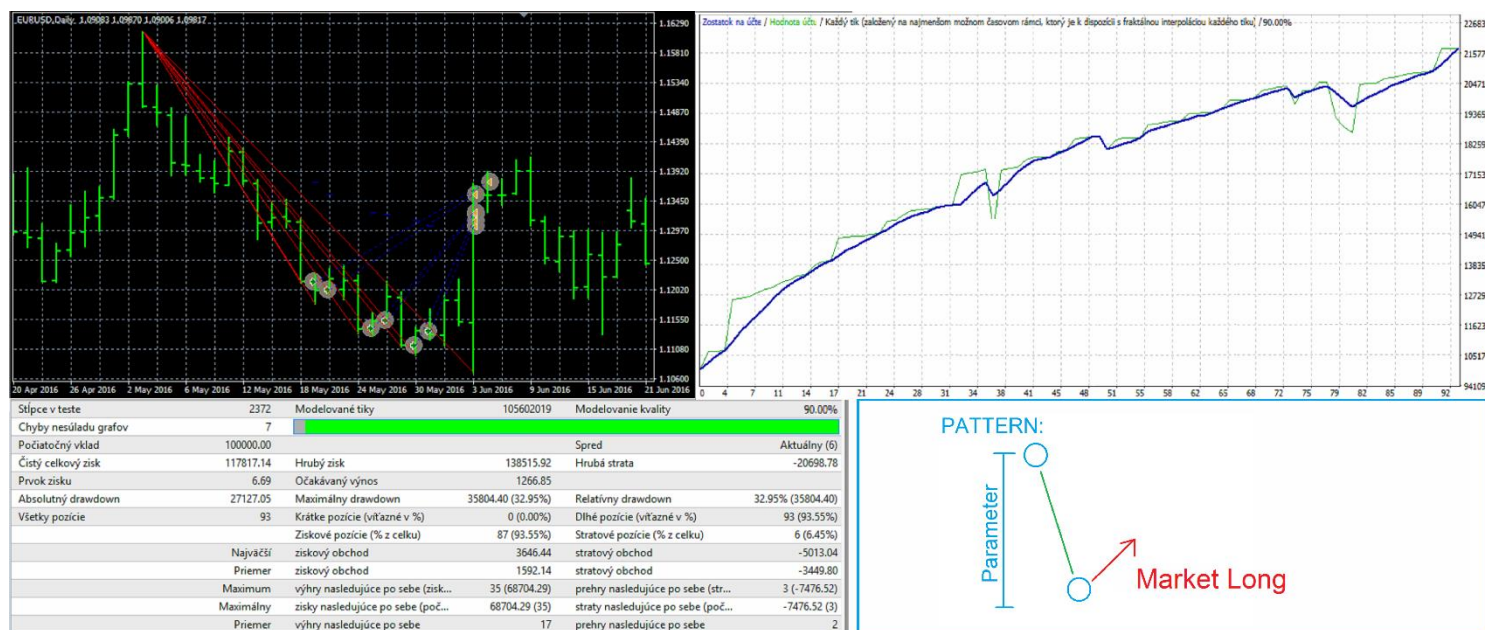
Na záver ukážem, že aj pri obchodných stratégiách platí, že v jednoduchosti je krása, na ďalšom obrázku možno vidieť pattern pozostávajúci len z dvoch uzlov, pričom jeho hlavnými parametrami je minimálna veľkosť, nastavenie indikátora a target zóny u obchodného príkazu. Test prebehol za obdobie posledných piatich rokov na EURUSD s prekvapivo vysokým ziskovým faktorom 11.50 pri 39 obchodoch.



Obrázok č. 51: UpTrend pattern

(Zdroj: Vlastné spracovanie)

Na tomto obrázku možno pozorovať výkonnosť inverznej verzie predošlého patternu s taktiež prekvapivo pozitívnym ziskovým faktorom 6.69 pri 93 obchodoch.



Obrázok č. 52: DownTrend pattern

(Zdroj: Vlastné spracovanie)

3.5 Vyhodnotenie

V tejto časti zhodnotím predvedený nástroj z môjho pohľadu, spomením jeho silné a slabé stránky v rámci jeho praktického využitia pre potenciálneho užívateľa.

3.5.1 Užívateľská prívetivosť

MetaTrader4 je populárnou, voľne dostupnou platformou s veľkou komunitou, preto možno povedať, že programovať takýto nástroj práve pre túto platformu je dobrým rozhodnutím, samotný kód je teda možné spustiť formou „plug and play“ kde si stačí iba stiahnuť kód a hneď ho je možné začať používať bez ďalších nutných úkonov. Rozličné patterny možno spustiť naraz, ako samostatné inštancie, v rámci jedného účtu. Dokopy tak môžu obchodníkovi tvoriť portfólio patternov ktoré by obchodoval. V prípade, že by sa ukázal niektorý z patternov nevýhodný, je možné jeho inštanciu jednoducho vypnúť a nahradiť iným nastavením, samotné prebiehajúce obchody tiež možno kontrolovať za behu a meniť ich cieľové hodnoty alebo ich ručne zatvárať.

Nevýhodou však môže byť práve to, že samotný nástroj nedisponuje UI ktoré by vizuálnejšou formou previedlo užívateľa nastavením si svojho patternu, jednotlivé nastavenia je nutné riešiť priamo v kóde, kde sa nastavujú hodnoty priamo do premenných, samozrejme človeku ktorý túto platformu pozná, nájdenie kľúčových nastavení nebude robiť problém, sú zoskupené vo vrchnej časti kódu a dostatočne okomentované poznámkami.

3.5.2 Robustnosť kódu

Samotný kód po dôkladnom testovaní nevykazuje žiadne náhodné havárie alebo zacyklenia, a to ani pri chybných nastaveniach, v takých prípadoch však možno očakávať, že buď nedôjde k nijakému nálezu alebo bude nájdený iný pattern, než by sme chceli. Pokiaľ je program správne nastavený, možno tvrdiť, že si plní svoju funkciu, jediné znepokojenie môže vzniknúť z kvality backtestov, z robustnosti a stability samotnej platformy MT4, ktorá je z môjho pohľadu značne orientovaná k bežným užívateľom, a teda si nie som istý stabilitou jej dlhodobého behu a funkčnosti, ako napríklad v prípade výpadkov spojenia. Z mojej skúsenosti bola platforma schopná bežať minimálne mesiac bez problémov, avšak po mesiaci samotný Windows začal robiť problémy (nečakane).

Výhodou je však to, že kód možno kedykoľvek reštartovať a nie je nutné sa obávať, že následkom krátkeho zastavenia, dôjde k porušeniu integrity stratégie alebo ohrozenia práve otvorených pozícií. To znamená, že možno reštartovať MetaTrader alebo aj celý počítač a následne spustiť algoritmus s rovnakým nastavením bez akýchkoľvek následkov.

Zjavnou nevýhodou je však robustnosť z pohľadu „enterprise“ požiadaviek, kde ak by ste chceli obchodovať s kapitálom na úrovni investičných spoločností, platforma MetaTrader by nebola najrozumnejšou voľbou, v takom prípade je vhodné migrovať know-how na robustnejšiu platformu, či už zo softvérového pohľadu ale aj hardvérového prevedenia, a to na úrovni serverových riešení.

3.5.3 Ekonomické zhodnotenie

Samotná platforma MetaTrader4 je voľne dostupnou, tak že nepredstavuje žiaden náklad pre potenciálneho užívateľa, otázkou je len to či jeho broker poskytuje možnosť ju využiť.

Vytvorenie kódu mi trvalo asi dva mesiace, to zahŕňa samotnú implementáciu ale aj vymyslenie princípu fungovania a odladenie chýb, možno tvrdiť, že MetaTrader je veľmi „user friendly“ platformou pre implementáciu svojich myšlienok, a to preto, že mnoho jeho funkcií je implicitne funkčných, tak ako by sa očakávalo, teda nie je nutné dlhé učenie sa ako samotná platforma funguje alebo riešenie mnohých explicitných nastavení ako je to napríklad u iných platforiem kde „low level control“ značne spomaľuje vývoj a implementáciu.

Obchodovateľných inštrumentov je obrovské množstvo, demonštráciou algoritmu som ukázal že možno nájsť profitabilné patterny a teda na nich založiť svoju tradingovú stratégiu, samotné profity sa samozrejme značne odvíjajú od zvolených patternov a teda je plne v rukách užívateľa aký bude jeho výsledok, mne sa však po relatívne krátkom testovaní podarilo dosiahnuť ziskového faktoru od 4 do 11 (pomer medzi celkovým ziskom a celkovou stratou). Na základe týchto testov na zvolených patternoch a inštrumentoch možno tvrdiť, že u najlepšieho patternu nám ziskové obchody vynesú až 11 násobok oproti tým stratovým, a v najmenej výkonnom prípade sa pohybujeme na štvornásobku.

Vo výsledku je mnou zvolené riešenie veľmi kapitálovo nenáročné a môže ho využiť ktokoľvek kto disponuje počítačom a internetom, jedinou bariérou pri používaní, je nutnosť, drobného, zorientovania sa v kóde, ktorý je v práci podrobne popísaný a časti kde sa definovali nutné parametre som vystavil a popísal aj v obrázkoch.

ZÁVER

V tejto práci som sa venoval problematike technických analýz a ich automatizovanému chodu. Konkrétna oblasť technickej analýzy, ktorou sa zaoberá táto práca sa nazýva rozpoznávanie patternov (vzorov). V tejto oblasti sa venujem predikciám vývoja ceny na základe správania v minulosti. Očakávam teda, že niektoré situácie sa na trhu cyklicky opakujú. Mojou taktikou je včasné zachytenie týchto cyklov a následné využitie týchto informácií k profitu. Po nájdení vhodných patternov začne pripadať v úvahu automatizácia v procese ich detekcie a exekúcie následných obchodných príkazov.

V tejto práci sa venujem práve samotnému nástroju, ktorý po zadaní vhodných parametrov dokáže automaticky tieto patterny detekovať a vyvodíť vopred definovanú akciu po ich nájdení. V praxi možno rozpoznávať viacero kategoricky veľmi odlišných patternov, a to patterny statické, tie sa sú zložené z predom definovanej postupnosti sviečok s drobnými toleranciami odchýlok. Druhú kategóriu patternov možno nazvať patterny dynamické, nimi sa práve zaoberám v tejto práci. Dynamický pattern sa na rozdiel od statického skladá z ľubovoľného množstva sviečok a teda rozhoduje výsledný vizuálny stav tejto časovej rady sviečok. Takéto patterny sú omnoho náročnejšie na algoitmické rozpoznávanie. Treba tiež prihliadať na to, že patterny sa neopakujú v 100%-nej zhode, ale majú drobné odlišnosti vo veľkostiach a tvaroch.

Algoritmus popisovaný v tejto práci dokáže identifikovať tieto patterny, filtrovať ich podľa požadovanej veľkosti a zadať následný obchodný príkaz v reálnom čase, Algoritmus je taktiež plne použiteľný na backtesting. Pre jeho inicializáciu stačí algoritmu zadať pattern v podobe hlavných uzlov, z ktorých sa má skladať, toleranciu odchýlok pri ich rozpoznávaní a obchodný príkaz ktorý sa má zadať pri nájdení patternu. Ďalšie dôležité parametre sú nastavenia podporného indikátora, ktorý identifikuje samotné ulzly v grafe a následne na nich sa hľadá pattern.

V práci som demonštroval aj samotné nasadenie algoritmu na kryptomene Bitcoin, ako aj na menovom páre EUR/USD. Možno pozorovať rozpoznanie patternov ktoré som si definoval a ich samotné zhodnotenie. Podobných stratégií možno vytvoriť nespočetne a možno ich testovať na ľubovoľnom instrumente. Samotný nástroj však nemusí byť silnou zbraňou, pokiaľ nie je v rukách skutočného obchodníka, ktorý vie ako optimálne nastaviť jednotlivé parametre. A to predovšetkým na základe reálnych znalostí o trhu a skutočných analýz. Avšak aj pre neskúseného užívateľa môže slúžiť ako nápomocný nástroj, najmä pri overovaní špekulatívnych hypotéz a pokusoch, bez toho, aby bolo nutné ručne hľadať v grafe pattern, je možné si ho rýchlo a bezstarostne otestovať v priebehu jednotiek minút.

V rámci budúcich vylepšení by bolo možné pridať možnú spoluprácu s voliteľnými indikátormi, ako podmienku pre otvorenie obchodu, príklad: Ak sa pattern nachádza 100pip nad kľavým priemerom, otvor obchod!

ZOZNAM POUŽITÝCH ZDROJOV

1. DOSTÁL, P. Advanced Decision Making in Business and Public Services. Brno: CERM, 2011. 168 s. ISBN 978-80-7204-747-5.
2. REJNUŠ, Oldřich. *Finanční trhy*. 3., rozš. vyd. Ostrava: Key Publishing, 2011. Ekonomie (Key Publishing). ISBN 978-80-7418-128-3.
3. HARTMAN, Ondřej. *Jak se stát forexovým obchodníkem: naučte se vydělávat na měnových trzích*. 2., rozš. vyd. Praha: FXstreet, c2014. ISBN 978-809-0441-835.
4. Objemy na Forexe [online]. [cit. 2020-02-20]. Dostupné z: <https://bmg-group.com/2008-financial-crisis/>
5. VIGNA, Paul a Michael CASEY. *The age of cryptocurrency: how bitcoin and the blockchain are challenging the global economic order*. New York: Picador, St. Martin's Press, 2016. ISBN 978-1-250-06563-6.
6. *Ako funguje Bitcoin* [online]. [cit. 2020-04-19]. Dostupné z: <https://www.youtube.com/watch?v=19jOJk30eQs&t=19s>
7. *Elliottove vlny* [online]. [cit. 2020-04-19]. Dostupné z: <https://forex-financeandtrading.com.ve/como-utilizar-la-onda-de-elliott-en-forex/>
8. *SMA* [online]. [cit. 2020-04-19]. Dostupné z: <https://forex-financeandtrading.com.ve/como-utilizar-la-onda-de-elliott-en-forex/>
9. Forex Market IFCM [online]. Copyright ©W [cit. 21.04.2020]. Dostupné z: https://www.ifcmarkets.com/pdf_files/forexbook/en/Forex_Trading_Strategies.pdf
10. Moving Average (MA) Definition. Investopedia: Sharper insight, better investing. [online]. Dostupné z: <https://www.investopedia.com/terms/m/movingaverage.asp>
11. CFD Broker | CFD Trading | Online Forex Trading | Forex Broker | IFCM UK [online]. Copyright © [cit. 21.04.2020]. Dostupné z: https://www.ifcm.co.uk/pdf_files/forexbook/en/IFCMarketsBook.pdf
12. Forex History FXGM [online]. [cit. 2020-04-19]. Dostupné z: <https://www.fxgm.com/trading-academy/forex-history/>
13. *Head and shoulders pattern* [online]. In: . [cit. 2020-04-21]. Dostupné z: <https://forexmt4ea.com/head-and-shoulders/head-shoulders-bearish/>
14. *Bitcoin Whitepaper* [online]. [cit. 2020-03-01]. Dostupné z: <https://bitcoin.org/bitcoin.pdf>

ZOZNAM POUŽITÝCH OBRÁZKOV

Obrázok č. 1: Porovnanie objemov na forexe s inými trhmi	15
Obrázok č. 2: Vývoj hodnoty futures kontraktu	16
Obrázok č. 3: Sviečkový graf ukážka	17
Obrázok č. 4: Kompozícia a význam rastúcej sviečky	18
Obrázok č. 5: Kompozícia a význam klesajúcej sviečky.....	19
Obrázok č. 6 Rozhranie obchodnej platformy xStation 5.....	21
Obrázok č. 7: Proces otvárania novej objednávky v xStation 5	22
Obrázok č. 8: Sledovanie aktualít v platforme	24
Obrázok č. 9: Sentiment trhu	24
Obrázok č. 10: Ilustrácia distribuovanej siete.....	26
Obrázok č. 11: Spracovanie transakcii „from pool to block“	27
Obrázok č. 12: Bitcoin na burze	28
Obrázok č. 13: Jednoduchý Kĺzavý Priemer	29
Obrázok č. 14: Bollingerové pásmo	31
Obrázok č. 15 Momentum Oscilátor.....	32
Obrázok č. 16 Rate of Change Oscilátor	33
Obrázok č. 17 Relative strength Index Oscilator	33
Obrázok č. 18 MACD Oscilátor	34
Obrázok č. 19: Znázornenie vzťahu vnútornej a reálnej ceny	35

Obrázok č. 20: Head and Shoulders pattern	37
Obrázok č. 21: Triangle pattern	37
Obrázok č. 22: Elliottove vlny	38
Obrázok č. 23: Quantopian	41
Obrázok č. 24: Quantconnect backtest výsledok	42
Obrázok č. 25: MetaTrader 4	43
Obrázok č. 26: MetaEditor.....	43
Obrázok č. 27: Príklady niektorých možných vzorov.	44
Obrázok č. 28: Vizuálna reprezentácia definovania uzlov v patterne.	45
Obrázok č. 29: Intervalová definícia uzlov pomocou offsetu.....	46
Obrázok č. 30: Meranie výšky patternu v „pip“ hodnotách	47
Obrázok č. 31: Indikátor ZigZag v MetaTrader 4	47
Obrázok č. 32: Princíp indikátoru ZigZag	48
Obrázok č. 33: Znázornenie postupného prekresľovania indikátoru ZigZag	50
Obrázok č. 34: Príklad patternu v praxi	52
Obrázok č. 35: Rozpoznávanie uzlov algoritmom ilustrovane	53
Obrázok č. 36: Porovnanie dvoch rôznych nastavení môjho indikátoru	53
Obrázok č. 37: Implementácia poľa uzlov	54
Obrázok č. 38: Hlavná procedúra podporného indikátora	55
Obrázok č. 39: Implementácia funkcie vyhodnocujúcej sviečky za uzly	56
Obrázok č. 40: Implementácia pridávania nových uzlov do poľa	57

Obrázok č. 41: Implementácia definície patternu	58
Obrázok č. 42: Implementácia vnútorného cyklu porovnávania uzlov indikátora s hľadaným patternom	60
Obrázok č. 43: Implementácia vykresľovania nájdeného patternu do grafu	61
Obrázok č. 44: Implementácia spínačového ovládania obchodných príkazov	62
Obrázok č. 45: Demonštrácia algoritmu na patterne Head&Shoulders	63
Obrázok č. 46: Pattern 70,0,100	63
Obrázok č. 47: Pattern „Bull Triangle“ 0,100,40,70,40	64
Obrázok č. 48: Pattern „Bull Double Top“ 0,80,30,100,30	64
Obrázok č. 49: Pattern „Bull Channel Brak“ 100,0,140,40	65
Obrázok č. 50: Pattern „Channel Brak“ 100,0,100,40	66
Obrázok č. 51: UpTrend pattern	66
Obrázok č. 52: DownTrend pattern	67